Neural network multi-model based method of fault diagnostics of actuators

Viktor Füvesi¹, Ernő Kovács²

¹,²University of Miskolc
¹Research Institute of Applied Earth Sciences, ¹Department of Research Instrumentation and Informatics
²Department of Electrical and Electronic Engineering
¹HU-3515 Miskolc-Egyetemváros, POB. 2
²HU-3515 Miskolc-Egyetemváros
¹fuvesi@afki.hu; ²elkke@uni-miskolc.hu

22nd International Symposium on Power Electronics,
Electrical Drives, Automation and Motion
June 18–20, 2014
Outlines

- Brief introduction of starter motor
- Measurement arrangement
  - Hardware, software, results
- NN Observers of the system
  - Structure of ANN
  - Training process
  - Model selection
- Model-based fault diagnoses
  - Fault generation
  - Active training process
  - Evaluation of the networks
Measurement arrangement

1 – starter motor
2 – clutch
3 – powder brake
4 – controller of the brake

- Control the break
- Measure different parameters
- 800 Hz sample frequency
Measurements

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Unit</th>
<th>Lower range</th>
<th>Upper range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Armature current (I)</td>
<td>A</td>
<td>0</td>
<td>300</td>
</tr>
<tr>
<td>Voltage of the battery (U)</td>
<td>V</td>
<td>9.5</td>
<td>13</td>
</tr>
<tr>
<td>Velocity of the motor (ω)</td>
<td>rpm</td>
<td>0</td>
<td>5000</td>
</tr>
<tr>
<td>Shaft torque (T_{shaft})</td>
<td>Nm</td>
<td>0</td>
<td>6</td>
</tr>
</tbody>
</table>

Constant load with different value

Different characteristic of load
Developed observers

"Black Box" model

Current

Transformation

Battery voltage

Estimation of velocity

"Black Box" model

Current

Transformation

Battery voltage

Estimation of shaft torque
Datasets

- Measurement data were formatted
  - Filtered with low-pass filter
  - Scaled and normalised to 0.1-0.9 interval
- Different datasets were developed for training and validating
Applied NN structure and dynamics

- Feedforward neural network was used
  - One hidden layer
  - Perceptron neuron structure
  - Elliott transfer function
  - Bias neurons

- MISO – Multi Input Single Output System
  - External dynamics was applied

\[ u(t-1) \]

\[ i(t-1) \]

\[ F \text{IR} \]
\[ o(t) = f[i(t-1), i(t-1-d), i(t-1-2d),..., i(t-1-md), u(t-1), u(t-1-d), u(t-1-2d),..., u(t-1-md)] \]

\[ A \text{RX} \]
\[ o(t) = f[i(t-1), i(t-1-d), i(t-1-2d),..., i(t-1-md), u(t-1), u(t-1-d), u(t-1-2d),..., u(t-1-md), y(t-1)] \]
Training process

1. Setting up the number of inputs (Changing from 1 to 5 by 1 during the simulation series)
2. Setting the training and validating datasets (Changing the size of delay between the regressors)
3. Setting up the number of hidden neurons (Set it to double of the inputs of NN)
4. Training the configured neural network
   - Initialization: Nguyen-Widrow method
   - Training algorithm: Rprop BP algorithm
   - Activation function: Elliott function
   - Maximal iteration number: 4000 epochs
5. Performance criteria:
   - MSE for actual training and validating sets
6. End of building ANN

- Delay among regressors (d)
- Number of inputs (m+1)
- 2, ..., 5
- 10, 20, 30, ..., 220
Selection process

Number of inputs \((m+1)\)

2, \ldots, 5

10, 20, 30, \ldots, 220

Delay among regressors \((d)\)

Results

88 FIR velocity observer model
88 ARX velocity observer model
88 FIR torque observer model
88 ARX torque observer model
Compare models

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y - o)^2 \]  
\[ AIC = n \cdot \log(MSE) + 2 \cdot (w + b) \]  
\[ BIC = n \cdot \log(MSE) + (w + b) \cdot \log(n) \]  
\[ FPE = n \cdot \log(MSE) + n \cdot \log\left(\frac{n + w + b}{n - w - b}\right) \]  

\[ PCC = \frac{\sum_{i=1}^{n} (y_i - \bar{y})(o_i - \bar{o})}{\sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2 \sqrt{\sum_{i=1}^{n} (o_i - \bar{o})^2}} \]
## Best NN models

<table>
<thead>
<tr>
<th>Model</th>
<th>Number of virtual inputs (m)</th>
<th>Number of delays among regressors</th>
<th>Velocity observer</th>
<th>Torque observer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>MSE for validation dataset</td>
<td>PCC for validation dataset</td>
</tr>
<tr>
<td>NN1</td>
<td>5</td>
<td>50</td>
<td>26598.72</td>
<td>0.992</td>
</tr>
<tr>
<td>NN2</td>
<td>5</td>
<td>60</td>
<td>32055.84</td>
<td>0.990</td>
</tr>
<tr>
<td>NN3</td>
<td>4</td>
<td>80</td>
<td>15769.38</td>
<td>0.995</td>
</tr>
<tr>
<td>NN4</td>
<td>5</td>
<td>110</td>
<td>37488.74</td>
<td>0.988</td>
</tr>
</tbody>
</table>

### Best neural network based models

\[
EF_{SC} = \sqrt{\text{norm}^2(\text{SC}_{\text{velocity}}) + \text{norm}^2(\text{SC}_{\text{torque}})} \tag{8}
\]
Model-based fault diagnostics

- Model-based fault detection tasks
  - Detect different additive faults in the analysed system

- Model-based fault diagnostic tasks
  - It is more than detection
  - Estimate the size of the fault

- Separation of the different faults
Structure of diagnoses system

- **F1** – Fault in voltage signal
- **F2** – Fault in current signal
- **F3** – Fault of velocity signal
- **F4** – Fault of torque signal

Diagram:
- Input: Voltage (U), Current (I)
- Output: Motor torque (T_m), Motor speed (ω_m), Sensor readings
- Observers: Velocity, Torque

Graph:
- Time (s) vs. F1 signal
- Time-axis: 0, 3650, 7299, 10948
- Signal-axis: 0.0, 0.2, 0.4

Equations:
- Speed (ω) = \(\frac{v}{r}\)
- Torque (T) = \(\frac{I}{r}\)

June 18–20, 2014 - SPEEDAM 2014
Faults in the system

- Fault of battery
- Problem in the connection between the starter and the accumulator
- Resistance of terminals suddenly rises
- Brake of the shaft of the motor
- Unexpected stalling of the shaft of the motor
- Malfunction of the valves

- F1 – Fault in voltage signal
- F2 – Fault in current signal
- F3 – Fault of velocity signal
- F4 – Fault of torque signal
Dataset generation

<table>
<thead>
<tr>
<th>PARAMETERS</th>
<th>TRAINING DATASET</th>
<th>VALIDATING DATASET</th>
<th>TEST DATASET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of dataset</td>
<td>1</td>
<td>1</td>
<td>240</td>
</tr>
<tr>
<td>Size of a dataset</td>
<td>231900</td>
<td>36490</td>
<td>7300</td>
</tr>
<tr>
<td>Number of fault in a dataset</td>
<td>72</td>
<td>16</td>
<td>3</td>
</tr>
</tbody>
</table>

- **Artificially generated faults**
  - Impulse type faults
  - Fault size varied from -30 to 30 %
Active training process

Setup main parameters of the NN (Number of inputs, Number of hidden layer, etc.)

Setup basic training and validation dataset

Start Training Process

Is it time to change the training dataset?

Yes

Run the network with actual training dataset and get actual MSE value

No

Select datapoints from training epoch, where the error is bigger than MSE of epoch

Generate new training dataset

Train the network using actual training dataset

Calculate MSE of validation dataset

Is the estimation of actual network better than last saved network?

Yes

Save network

No

Is it the end of training?

No

End Training Process

Yes

June 18÷20, 2014 - SPEEDAM 2014
Active training process

[Graph showing comparison between traditional and developed methods for different input numbers and training processes (Incremental, Batch, RProp, QProp)].

- Traditional method
- Developed method

Time of training process [min]

Number of inputs [no]
Active training process

![Graphs showing Active training process](image-url)
Evaluation of the networks

$ERA = \frac{RE}{RE + ME + BE}$

RE: Recognized edges
ME: Missed edges
BE: Bad edges
Estimation of fault size

Graphs showing the mean and standard deviation of generated fault sizes for different faults (F1, F2, F3, F4). The x-axis represents the nominal size of generated faults in percentage, while the y-axis shows the percentage of mean and standard deviation.
C/C++ programs were developed
- Pre- and post processing the data
- Train and evaluate networks
- Make diagrams

Used software
- Fedora 18, GCC, Code::blocks

Used libraries
- Fann – leenissen.dk/fann/wp/
- MathGL – mathgl.sourceforge.net/
- Armadillo – arma.sourceforge.net/
- GSL – www.gnu.org/software/gsl/
Different NN-based observers were developed based on measurements

Additive, impulse type faults were artificially generated

Multi model-based fault diagnoses were performed

- Active training process were used
- Complex evaluation process to find the best result
Acknowledgement

The research was partially carried out in the framework of the Center of Excellence of Sustainable Resource Management at the University of Miskolc.

This paper could be done with the support of TÁMOP-4.1.1.C-12/1/KONV-2012-0002: "Cooperation between higher education, research institutes and automotive industry".

The Project is supported by the Hungarian Government and co-financed by the European Social Fund.
References


Thank you for your kind attention!