List of sensor fault detection methods

Georg Jäger

12. November 2014

Zusammenfassung

This document is meant to list up different methods for sensor fault detection. Furthermore, all methods should be classified whether they can handle specific fault types or not. As we assume the sensor signal as the only input of the fault detection methods, another classification is done concerning the dynamic of the input signal. This classification is done by deciding whether a method is able to detect a specific fault type in a high/middle/low dynamic signal.

Inhaltsverzeichnis

1 Preparation

1.1 Fault model

We want to classify sensor fault detection methods concerning specific fault types. The first step is to introduce the underlaying fault model. The model used for this analysis was investigated by Sebastian Zug [1].

1.2 Process model

2 Methods for sensor fault detection

Sensor fault detection was addressed by many scientific publications and papers before. Therefore a lot of different approaches are available. As the

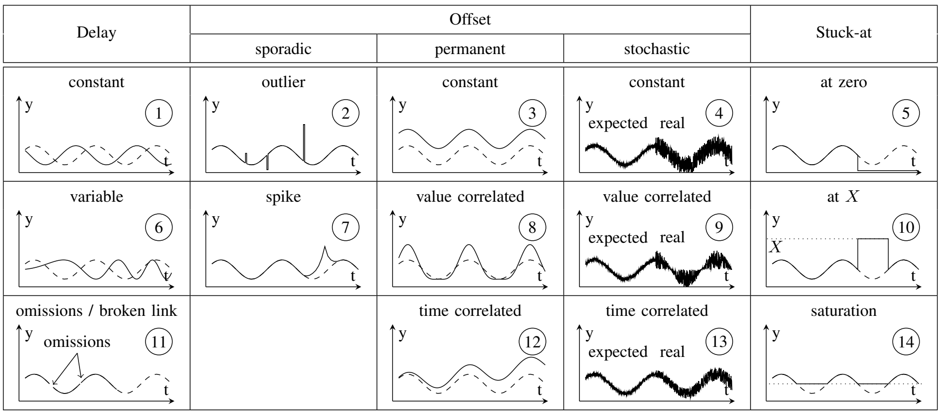

Abbildung 1: Faultmodel investigated by Sebastian Zug

primary focus of this documentation is on the framework, we only introduce and analyze some before selected data-driven techniques on detecting sensor faults. The input of every detection method is a signal representing the observations of an sensor over time. Beside this, we assume only one-dimensional sensors, e.g distance sensors. As the database will be of the same dimension as the input of the methods, the dimension will be 2 (sensor data and time).

Furthermore, we want to provide a kind of categorization of this data-driven methods: There are three different types of data-driven methods. At first, the fault detection techniques based on pattern recognition. This type of methods analyze the input data in order to find specified patterns, such as fault patterns. Prominent examples of this category are neural networks like *Time-Delay neural networks* and also classifications-methods like nearest-neighbor-classification.

Another type of detection techniques are using time-redundancy for residual or symptom generation. This methods tries to find redundant information in a time-series, such as the average which must be in a certain interval. If this information is deviating from normal values, sensor faults are detected. Examples of this category are *Limit Checking* on the gradient of a signal or an auto-correlation-analysis.

The last type are using models of the underlaying process. By predicting and comparing the sensor observations, they can generate a residual. The process model can be obtained by different approaches. As an example, one could use recurrent neural networks (Elman-networks, NARX) in order to „learn “the model. Another possibility is to determine the transfer function of the system.

2.1 Detection methods

2.2 Recognizing patterns of sensor faults

Multi-Layer-Perceptron
Time-Delay neural network
Wavelet-Analysis
Hidden-Markov-Models
Support-Vector-Machines
Fuzzy-Classifier
Nearest-Neighbor-Classification

2.3 Using time-redundancy for residual/symptom generation

Gradient-Checking
Average-Checking
Variance-Checking
Auto-correlation-analysis
Fourier-Analysis
Spectrum-Analysis
PCA (AANN)

2.4 Using process models for residual/symptom generation

MLP's
Recurrent neural network (Elman-Network, NARX)
State-Observer
HMM??
Method of (extended) least squares
Transfer-Functions(DGL)

Literatur

[1] Sebastian Ernst Zug. *Architektur für verteilte, fehlertolerante Sensor-Aktor-Systeme.* PhD thesis, Otto-von-Guericke-Universität Magdeburg, 2011.

Tabelle 1: Summary of all detection methods

Name	Features	Design-Time	Run-Time
1. *Limit Checking*	Gradient, Average, NLPCA-based residual,...	Find best feature and determine correct threshold	Calculate feature check threshold
2. *Adaptive Thresholds*	Gradient, Average, NLPCA-based residual,...	Find best feature and determing threshold function	Calculate feature calculate current threshold check threshold
3. *Multi-Layer-Perceptron*	Everything is possible	Get Database Determine best features Train MLP	Calculate feature Determine output of MLP
4. *State Observer*	ODE describing the system	Determine ODE	calculate state-space using ODE
5. *Time-Delay Neural Network*	Everything is possible	Get Database Determine best features Train NN	Calculate features transfer to running window determine output of NN
6. *Recurrent Neural Network*	Everything is possible	Get Database Determine best features Train NN	Calculate features determine output of NN generate residual by comparing output of NN with real sensor data
7. *Hidden Markov Model*	Possibilities of every state	Determine relevant states Calculate probabilities of transision	Update HMM-States
8. *Wavelet-Analysis*	-	Determine correct Wavelets	Wavelet-Analysis needs a convolution
9. *Fourier-Analysis*	Calculate running window of signal	determine normal frequencies and amplitudes	online fourier-transformation
10. *Auto-Correlation-Analysis*	-	determine normal autocorrelation of signal	requires convolution of signal

Tabelle 2: Classification: „Limit-checking of the signals gradient “

	High dynamic	Normal dynamic	Low dynamic
1. Outlier	not appropriate	OK	Well