Data-driven Detection and Identification of Undesirable Events in Subsea Oil Wells

Chrisander Brønstad
University South-Eastern Norway
email: c.bronstad94@gmail.com

Sergio L. Netto
Federal University of Rio de Janeiro
email: sergioln@smt.ufrj.br

Antonio L.L Ramos
University of South-Eastern Norway
email: antonior@usn.no

About the author

- MSc. Systems Engineering with Embedded Systems from the University of South-Eastern Norway (USN)
- Software Developer in Asker, Norway.

Overview

- Introduction
- System Framework
- Experiments and Results
- Conclusion

Oil & Gas Industry:

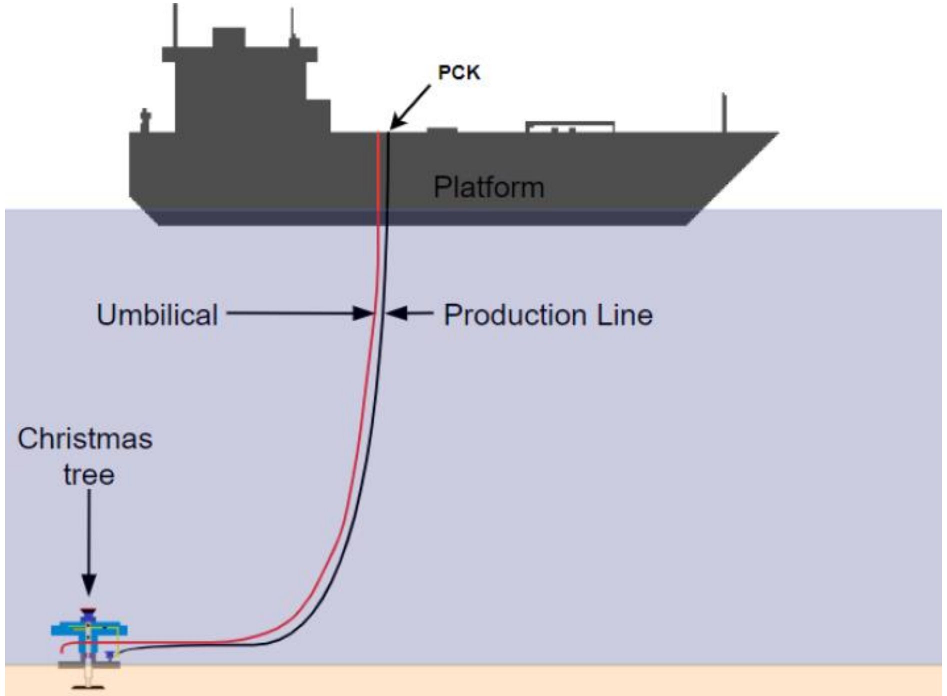
- Loss of 20 billion dollars every year due to abnormal events.
- Abnormal Event Management (AEM).
- The 3W dataset by Petrobras.

Condition-Based Monitoring (CBM) systems:

- Strategy to monitor and identify conditions of a process or machinery.
- A smart system making decisions without human interactions.
- Reduce cost and increase efficiency.

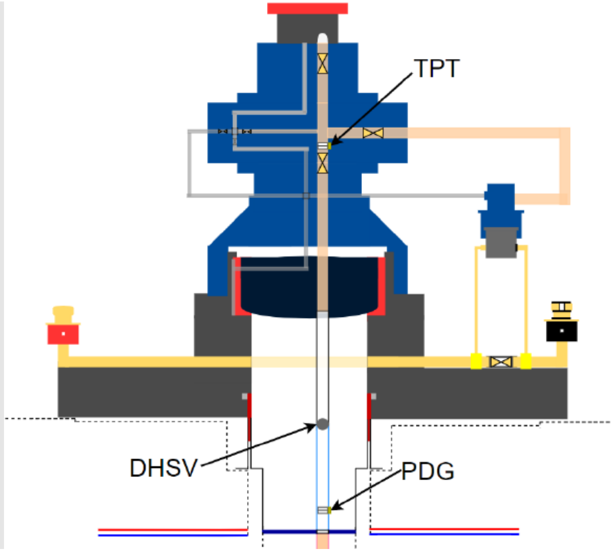
Introduction

Offshore oil wells



Monitored Variables/Sensors:

- Pressure at the Permanent Downhole Gauge (PDG).
- Pressure at the Temperature and Pressure Transducer (TPT).
- Temperature and the TPT.
- Pressure upstream of the Production Choke Valve (PCK).
- Temperature downstream of the PCK.
- Downhole Safety Valve (DHSV) – Closure Mechanism.



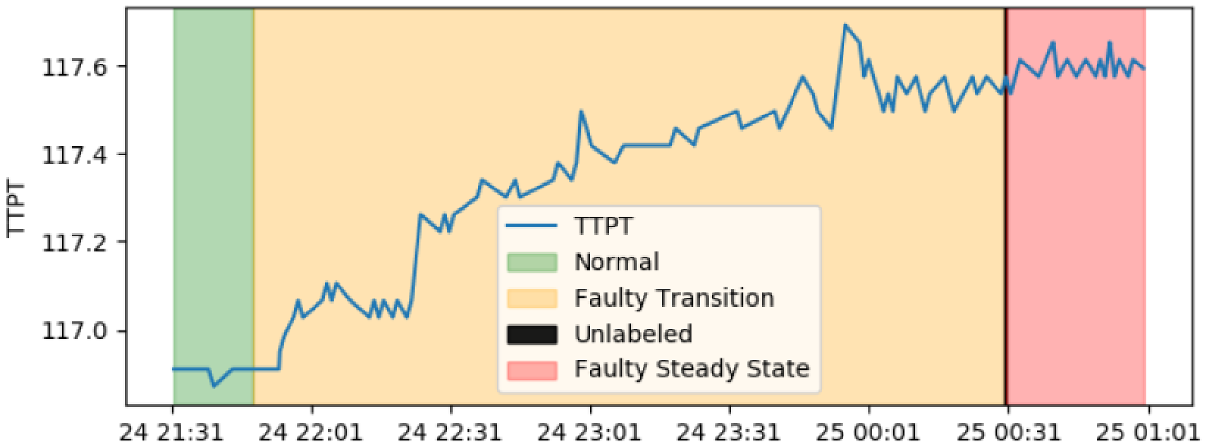
3W dataset

- A data set released by Petrobras.
- Real, simulated and hand-drawn instances.
- Eight real undesirable events:
  - 1. Abrupt increase of BSW.
  - 2. Spurious closure of DSHV.
  - 3. Severe slugging.
  - 4. Flow instability.
  - 5. Rapid productivity loss.
  - 6. Quick restriction in PCK.
  - 7. Scaling in PCK.
  - 8. Hydrate in production line.

Quantitative relation of the instances in the 3W dataset.

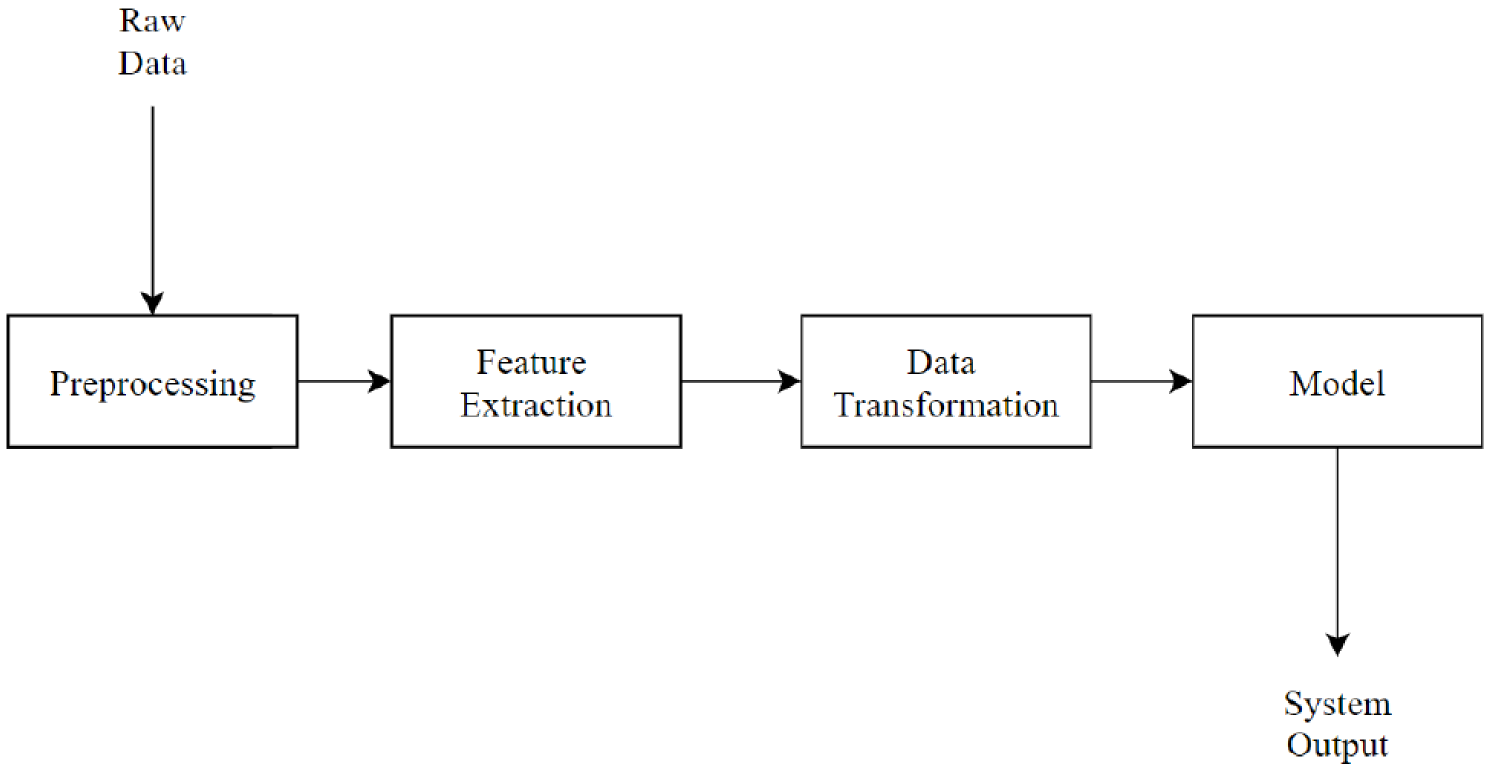
Type of Event	Real Instances	Simulated Instances	Hand-Drawn Instances	Total Instances
0. Normal	597	0	0	597
1. Abrupt Increase of BSW	5	114	10	129
2. Spurious Closure of DHSV	22	16	0	38
3. Severe Slugging	32	74	0	106
4. Flow Instability	344	0	0	344
5. Rapid Productivity Loss	12	439	0	451
6. Quick Restriction in PCK	6	215	0	221
7. Scaling in PCK	4	0	10	14
8. Hydrate in Production Line	3	81	0	84
Total	1025	939	20	1984

Real instance of ‘Abrupt increase of BSW’ showing the temperature at TPT



3

System Framework



1. Preprocessing

1. Split data into train and test (70/30)
2. Clean before feature extraction

Quantitative relation between the training and test set.

Type of event	Train Instances	Test Instances	Total Instances
0. Normal	418	179	597
1. Abrupt Increase of BSW	84	35	119
2. Spurious Closure of DHSV	27	11	38
3. Severe Slugging	74	32	106
4. Flow Instability	241	103	344
5. Rapid Productivity Loss	316	135	451
6. Quick Restriction in PCK	155	66	221
8. Hydrate in Production Line	59	25	84
Total	1374	586	1960

Total amount of Nan values and unlabeled observations for each event in the dataset.

Type of event	NaN Values	Unlabeled Observations
0. Normal	17,537,620	0
1. Abrupt Increase of BSW	26,729,233	1,019
2. Spurious Closure of DHSV	2,071,568	1,026
3. Severe Slugging	13,368,136	0
4. Flow Instability	4,289,933	0
5. Rapid Productivity Loss	38,948,805	1,461
6. Quick Restriction in PCK	23,286,058	622
8. Hydrate in Production Line	6,742,972	369
Total	132,974,325	4,497

2. Feature Extraction

Hyperparameters:

- Window size ***N***.
- Step size ***s***.

Nine statistical features:

- Mean and standard deviation.
- Skewness and Kurtosis.
- 5-number summary:
  - Minimum, median and maximum.
  - Lower and upper quartiles.

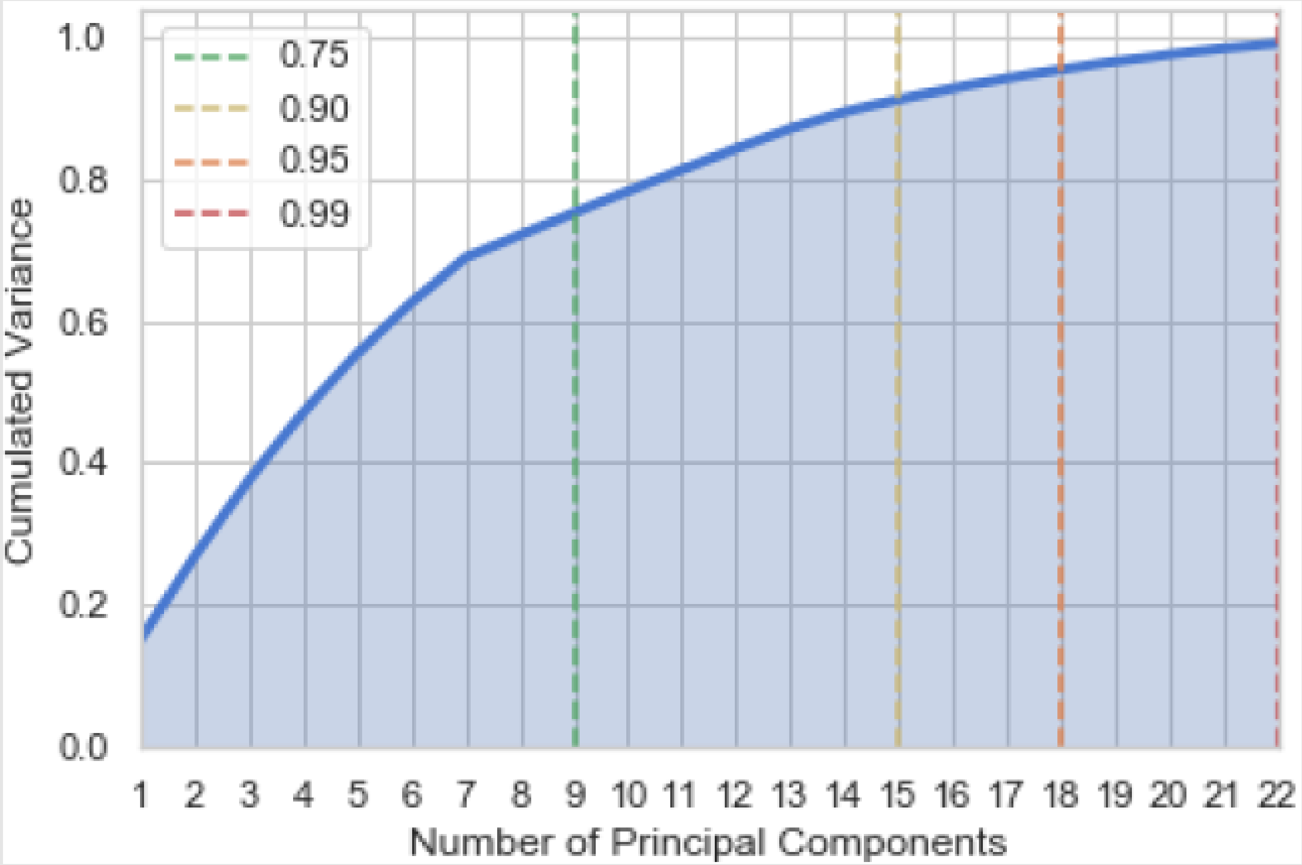
3. Data Transformation

Hyperparameter:

- PCA threshold τ.

Principal Component Analysis (PCA) :

Reducing the 72-dimensional data (eight sensors and nine features).



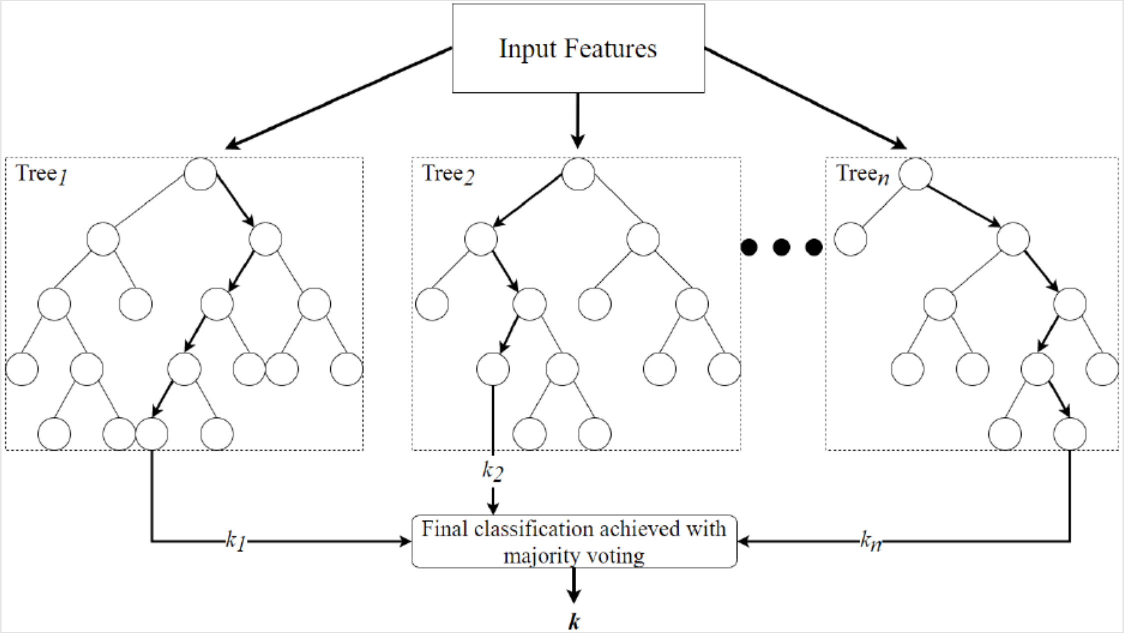
4. Classification Modeling

Hyperparameters:

- Max tree depth ***d***.
- Number of estimator ***E***.

Random Forests:

- Simple to train and deploy.
- It can handle small datasets.



Experiments and Results

Experiment 1:

Fault versus Normal Operation.

- Discriminate between faulty and normal operations.
- Applying multiple binary classifiers, one for each fault.
- The classifiers are fit on their specified fault and class 0 (normal operation).

Experiment 2:

Fault versus Not Fault.

- Discriminate between a specified fault and everything that does not belong to that certain fault.
- Applying multiple binary classifiers, one for each fault.
- The classifiers are fit on all classes.

Experiment 1: Fault versus Normal Operation

Overall Test Accuracy.

Fault	Window Size	Type	Transitional ACC	Overall ACC
Class 1	900	Real	0.214	0.989
		Simulated	0.999	0.999
Class 2	400	Real	0.986	0.999
		Simulated	0.996	0.996
Class 3	300	Real	0.998	0.999
		Simulated	0.998	0.998
Class 4	300	Real	0.967	0.986
		Simulated	-	-
Class 5	300	Real	0.888	0.992
		Simulated	0.993	0.993
Class 6	300	Real	0.972	0.999
		Simulated	0.951	0.951
Class 8	1200	Real	0.892	0.999
		Simulated	0.956	0.956

Test Accuracy for each transitional state.

Fault	Window Size	Type	Normal (Class 0)	Initial Normal	Transient State	Steady State
Class 1	900	Real	1.000	0.779	0.098	0.011
		Simulated	-	0.999	0.999	0.999
Class 2	400	Real	0.999	0.952	0.996	1.000
		Simulated	-	1.000	0.970	1.000
Class 3	300	Real	0.999	-	-	0.998
		Simulated	-	-	-	0.998
Class 4	300	Real	0.990	-	-	0.967
		Simulated	-	-	-	-
Class 5	300	Real	0.999	0.514	0.904	-
		Simulated	-	0.134	0.999	1.000
Class 6	300	Real	0.999	0.981	0.882	1.000
		Simulated	-	0.876	0.955	0.956
Class 8	1200	Real	0.999	0.000	1.000	1.000
		Simulated	-	0.258	0.996	0.999

Experiment 2: Fault versus Not Fault

Overall Test Accuracy.

Fault	Window Size	Type	Transitional ACC	Overall ACC
Class 1	900	Real	0.213	0.992
		Simulated	0.999	0.999
Class 2	400	Real	0.991	0.999
		Simulated	0.862	0.997
Class 3	300	Real	0.995	0.980
		Simulated	0.936	0.990
Class 4	300	Real	0.977	0.985
		Simulated	-	1.000
Class 5	300	Real	0.373	0.966
		Simulated	0.993	0.996
Class 6	300	Real	0.868	0.999
		Simulated	0.925	0.987
Class 8	1200	Real	0.892	0.999
		Simulated	0.972	0.982

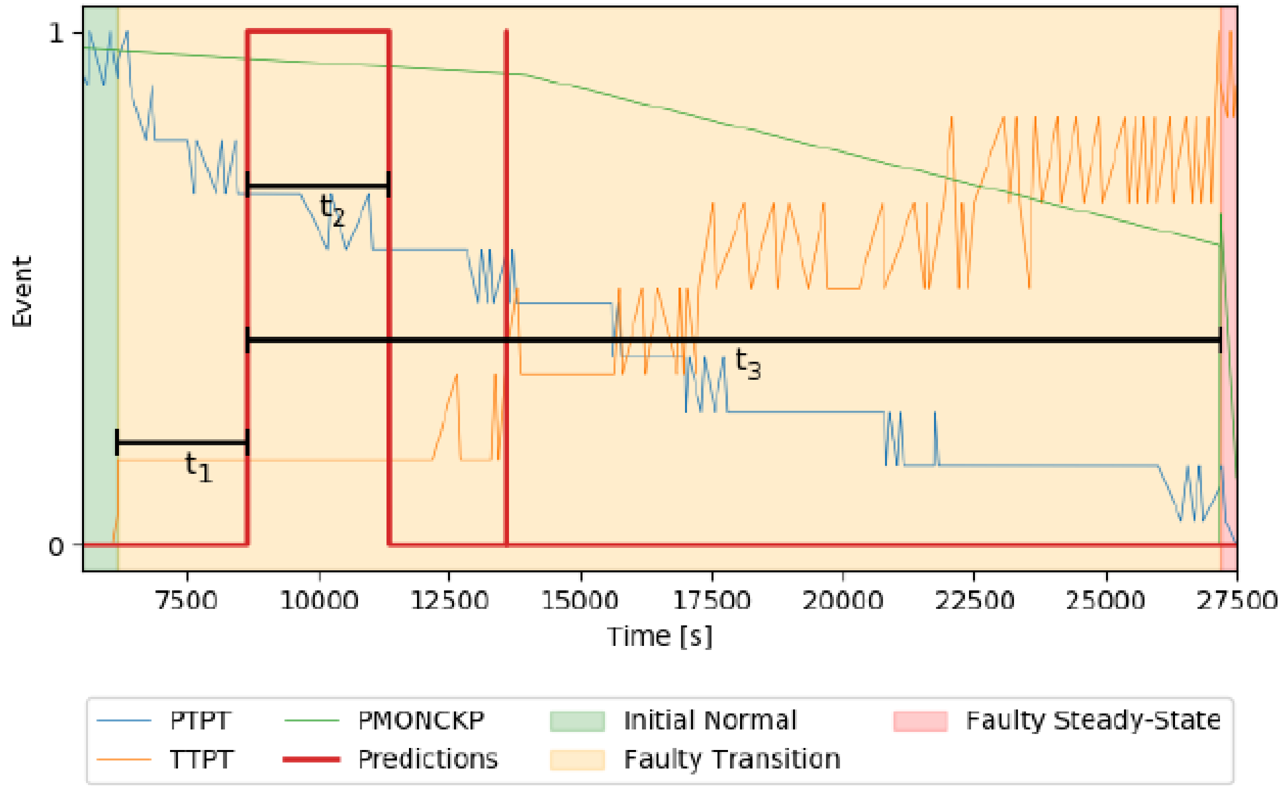
Test Accuracy for each transitional state.

Fault	Window Size	Type	Not Fault (Class 0)	Initial Normal	Transient State	Steady State
Class 1	900	Real	1.000	0.779	0.098	0.001
		Simulated	0.999	0.999	0.999	0.999
Class 2	400	Real	0.999	0.970	0.998	1.000
		Simulated	0.999	1.000	0.821	0.848
Class 3	300	Real	0.979	-	-	0.995
		Simulated	0.998	-	-	0.936
Class 4	300	Real	0.987	-	-	0.977
		Simulated	1.000	-	-	-
Class 5	300	Real	0.999	0.396	0.372	-
		Simulated	0.997	0.103	0.999	0.999
Class 6	300	Real	1.000	0.997	0.384	0.150
		Simulated	0.999	0.813	0.936	0.930
Class 8	1200	Real	0.999	0.000	1.000	1.000
		Simulated	0.983	0.939	0.961	0.999

Assessment as a CBM system

- System efficiency and reliability evaluation: How fast the system can detect an incoming failure (*t1*), the amount of consecutive correct predictions after the time of detection (*t2*), and how long time is left to prevent the incoming failure (*t3*).

A real instance of ‘Class 1’ predicted sample-wise.



System Efficiency and Reliability

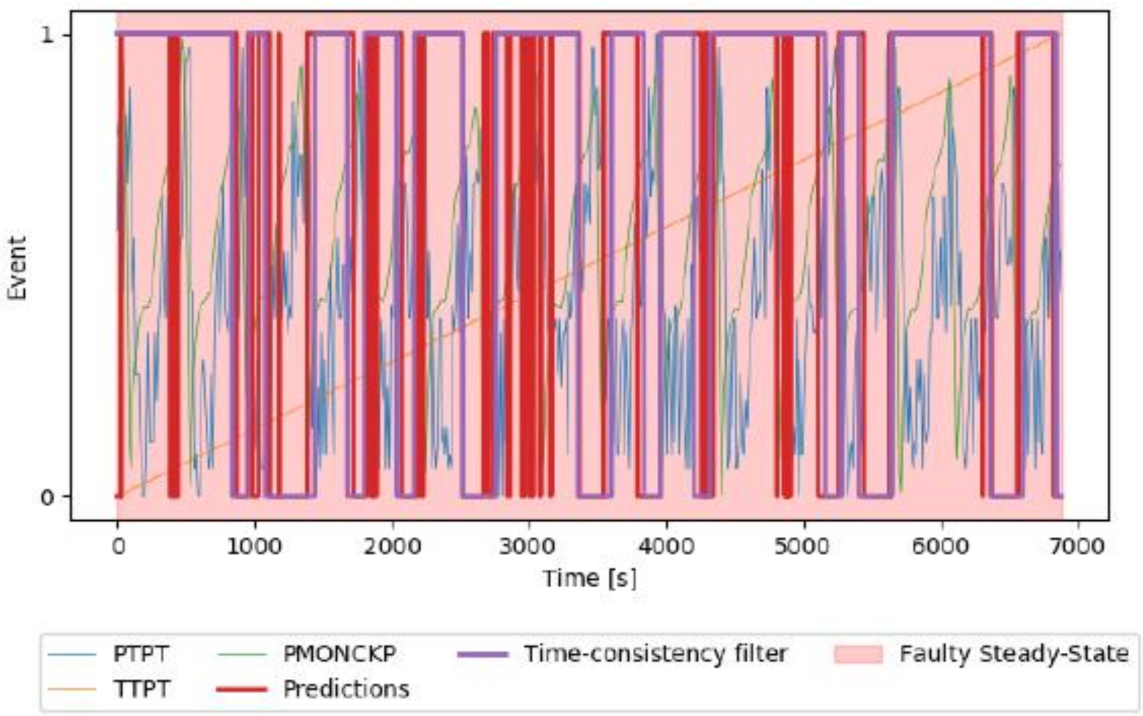
Fault versus normal operation			
Fault	t1 [s]	t2 [s]	t3 [s]
Class 1	2463.0 (11.73%)	2710.0 (12.90%)	18530.0 (88.26%)
Class 2	15.5 (0.33%)	4700.1 (99.67%)	4700.1 (99.67%)
Class 5	564.2 (1.06%)	41879.0 (78.87%)	52533.7 (98.93%)
Class 6	73.5 (11.87%)	546.0 (88.20%)	546.0 (88.20%)
Class 8	1.0 (0.00%)	20078.0 (100%)	20078.0 (100%)

Fault versus not fault			
Fault	t1 [s]	t2 [s]	t3 [s]
Class 1	2463.0 (11.73%)	2707.0 (12.90%)	18530.0 (88.26%)
Class 2	9.3 (0.19%)	4706.4 (99.81%)	4706.4 (99.81%)
Class 5	1.0 (0.00%)	822.3 (1.55%)	53097.3 (99.9%)
Class 6	318.5 (51.5%)	237.5 (38.4%)	300.5(48.54%)
Class 8	1.0 (0.00%)	20078.0 (100%)	20078.0 (100%)

Discussion

Time consistency filter

- A strategy to reduce inconsistency and fluctuating system classifications.
- A simple filter that evaluates the last 120 samples and filters out the class with the fewest output classifications.



Conclusion

- This work has successfully built a complete CBM system and applied complex machine learning tools, which are the current state-of-the-art.
- The system can extract features, reduce dimensionality, and classify with the popular random forest algorithm.
- Both classification scenarios have shown that it can not only detect failures but also anticipating the incoming failures, with an overall accuracy of 90%.
- Introduced a “time-consistency filter” to reduce inconsistent system classifications.