CentraleSupélec

Remaining useful life prediction with imprecise and partial information: An evidential hidden Markov model-based approach

Zhiguo Zeng （曾志国）

Assistant Professor
Chaire on Risk and Resilience of Complex Systems

CentraleSupélec, Université Paris-Saclay

zhiguo.zeng@centralesupelec.fr

Collaborate with PhD student Mr. Tangfan Xiahou and Prof. Yu Liu from UESTC, China.

28/11/2019

université PARIS-SACLAY

Contents

CentraleSupélec

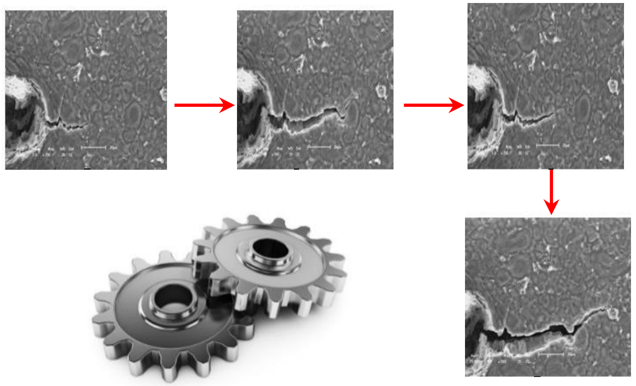
université PARIS-SACLAY

Research Background

Cutter wearing process

Blade creeping process

Metal oxidation process



Gear crack process

Engineering systems are subject to degradation because of workload and Environment

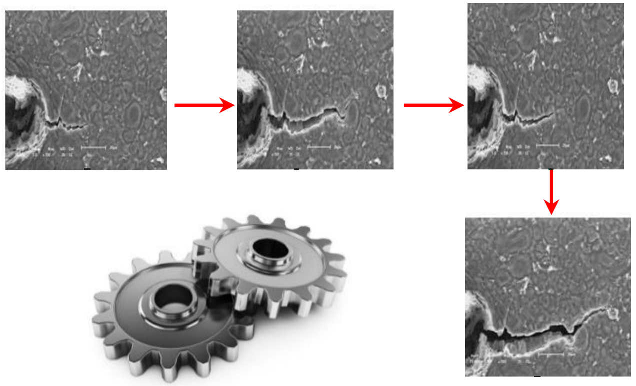
Health Prognostics, as one of the major tasks in Condition-Based Maintenance (CBM), aims to predict the remaining useful life (RUL) of machinery based on the historical and condition monitoring information.

Research Background

Cutter wearing process

Blade creeping process

Metal oxidation process



Gear crack process

Engineering systems are subject to degradation because of workload and Environment

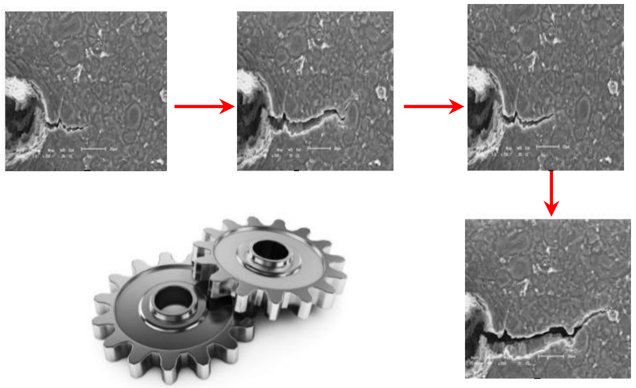
Health Prognostics, as one of the major tasks in Condition-Based Maintenance (CBM), aims to predict the remaining useful life (RUL) of machinery based on the historical and condition monitoring information.

Research Background

Cutter wearing process

Blade creeping process

Metal oxidation process



Gear crack process

Engineering systems are subject to degradation because of workload and Environment

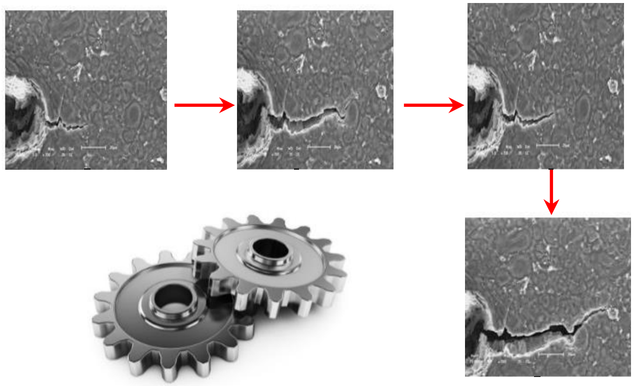
Health Prognostics, as one of the major tasks in Condition-Based Maintenance (CBM), aims to predict the remaining useful life (RUL) of machinery based on the historical and condition monitoring information.

Research Background

Cutter wearing process

Blade creeping process

Metal oxidation process



Gear crack process

Engineering systems are subject to degradation because of workload and Environment

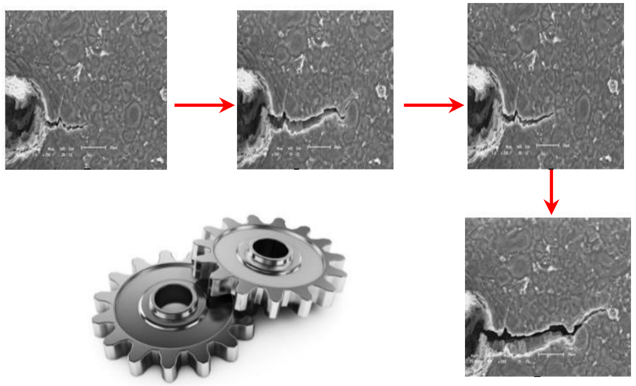
Health Prognostics, as one of the major tasks in Condition-Based Maintenance (CBM), aims to predict the remaining useful life (RUL) of machinery based on the historical and condition monitoring information.

Research Background

Cutter wearing process

Blade creeping process

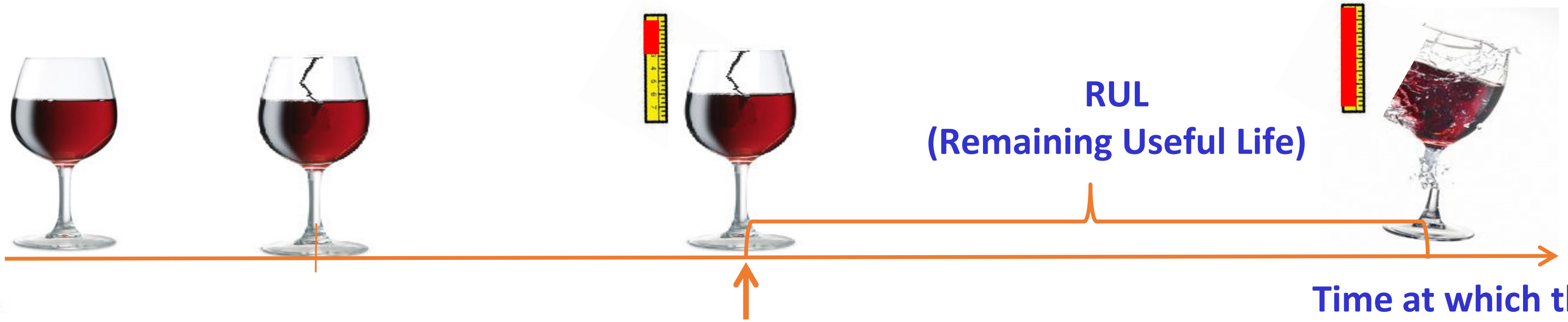
Metal oxidation process



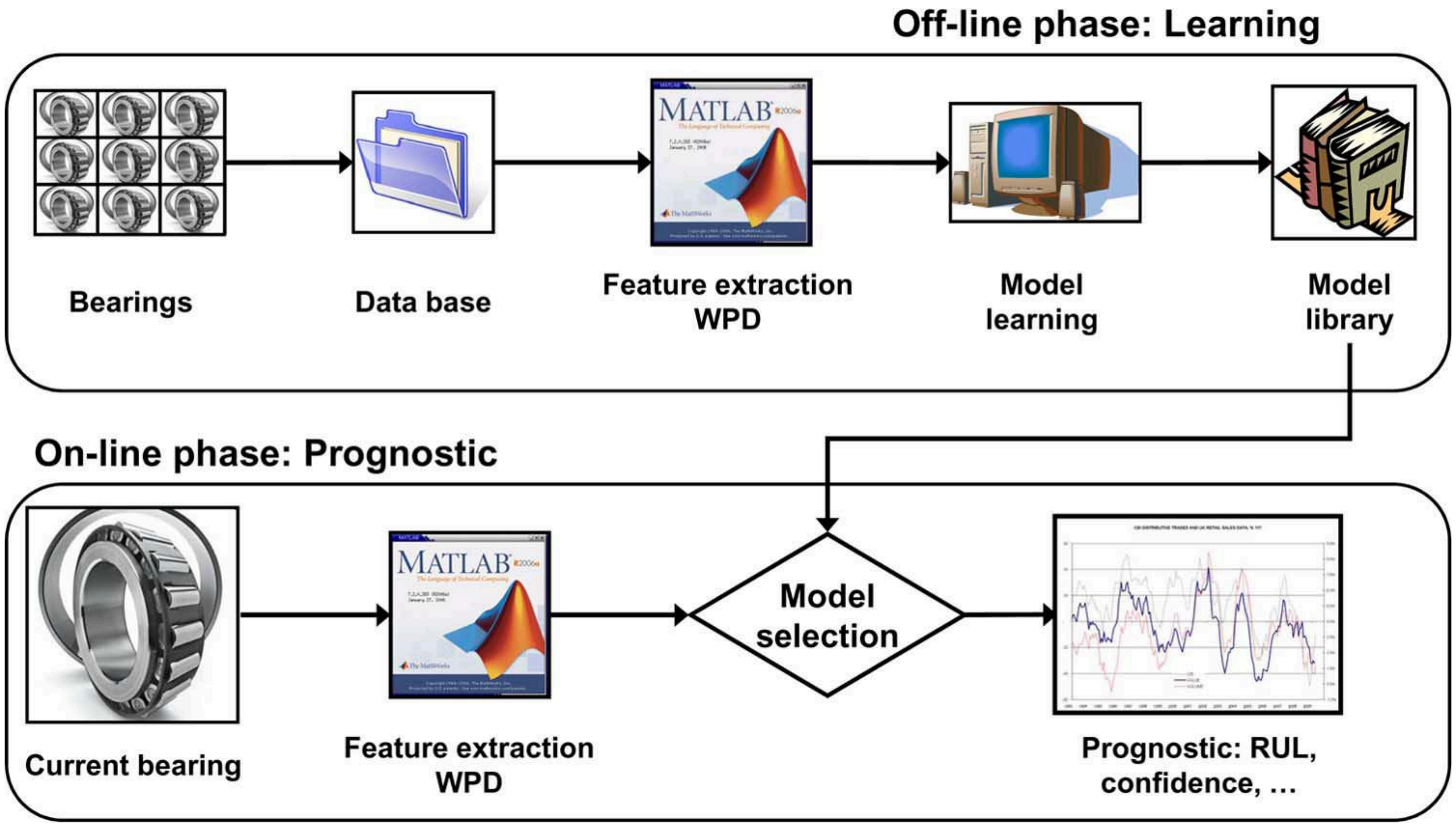
Gear crack process

Engineering systems are subject to degradation because of workload and Environment

Health Prognostics, as one of the major tasks in Condition-Based Maintenance (CBM), aims to predict the remaining useful life (RUL) of machinery based on the historical and condition monitoring information.



Research Background



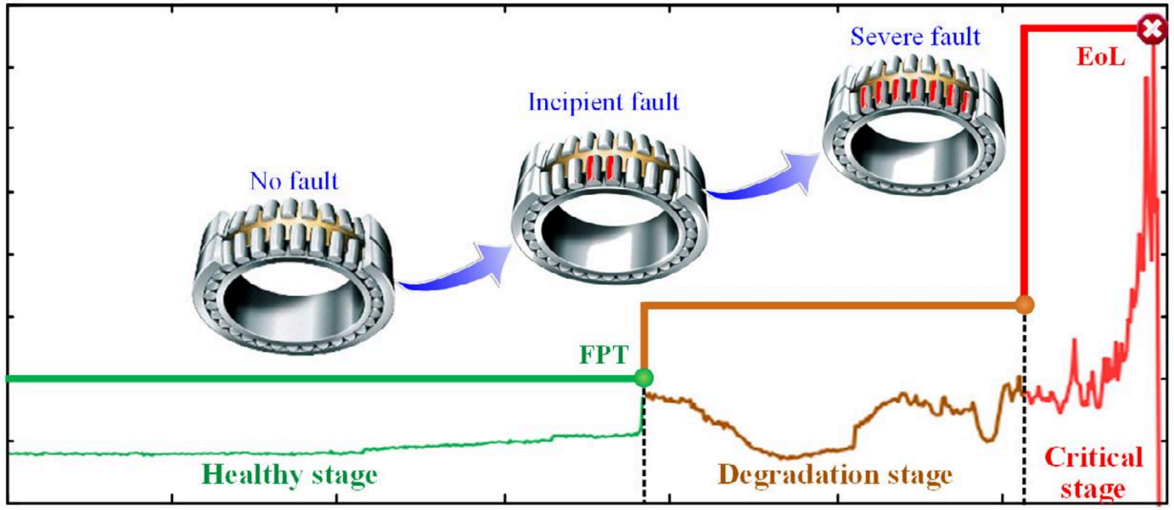
Procedures of Remaining Useful Life Prediction

Traditional RUL prediction methods cannot fully utilize the imprecise and partial knowledge/ information during the usage of the systems.

CentraleSupélec

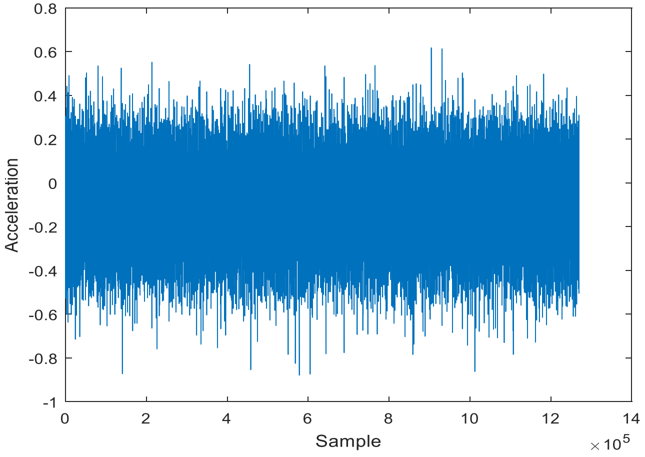
universite PARIS-SACLAY

Research Background

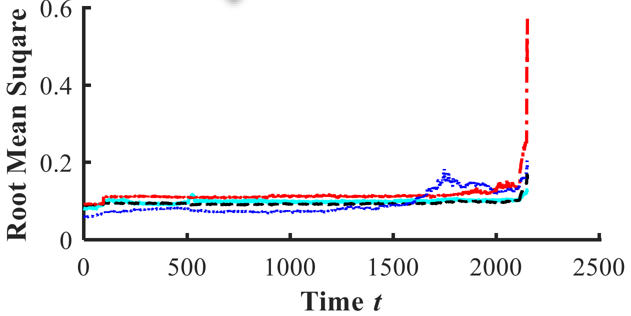


Multi-State Division

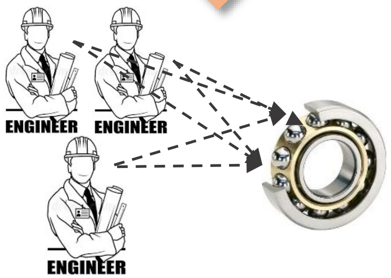
Condition Monitoring

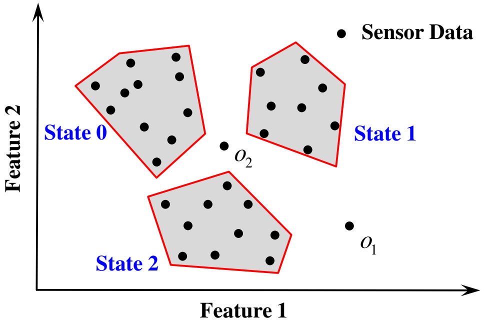


Feature Extraction



Clustering , Learning

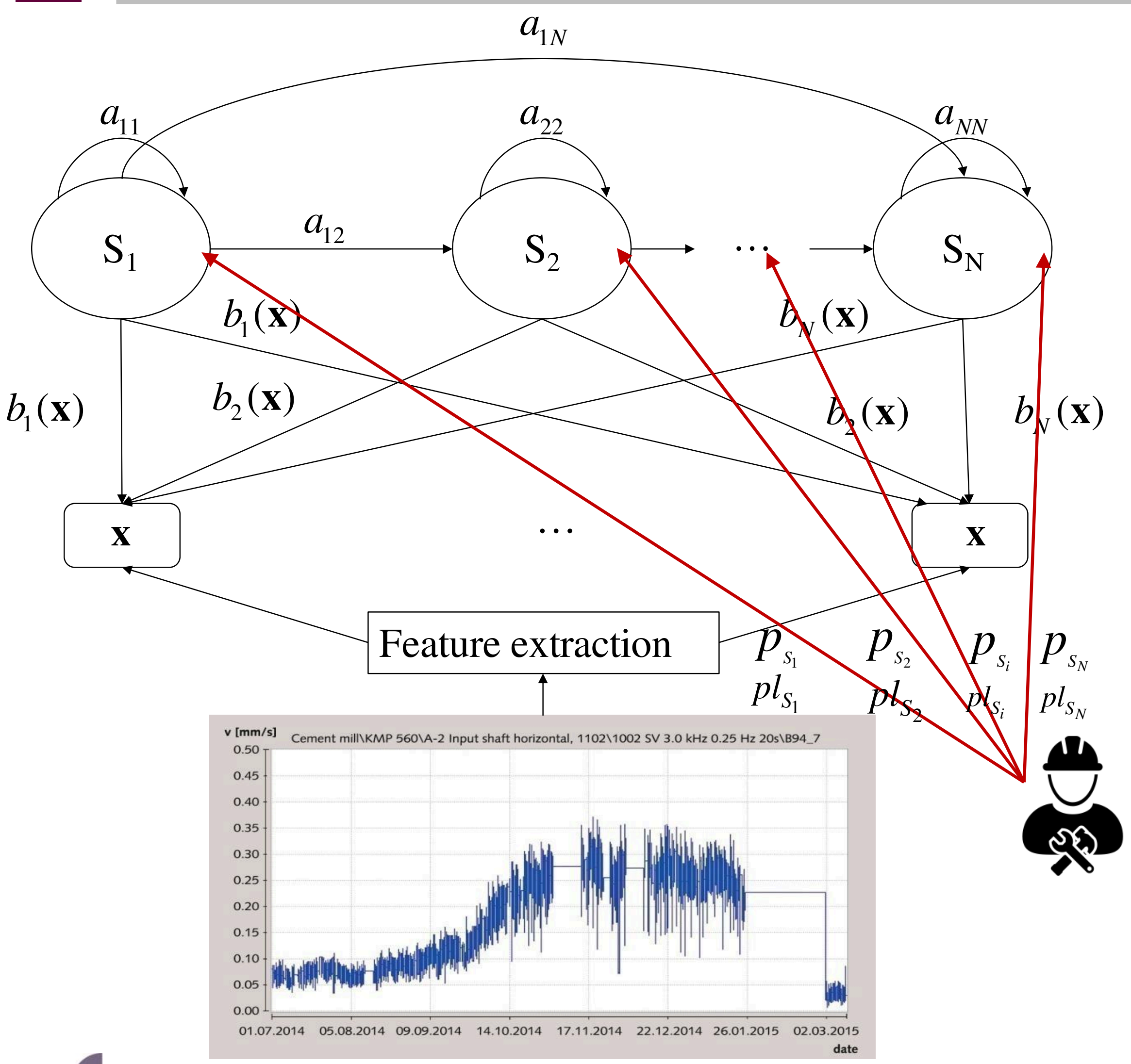




Combine

Experts' Partial Knowledge ⟷ Condition Monitoring Data

Research Background



Condition monitoring data

Gaussian Mixture Hidden Markov Model（GM-HMM）+ Expert knowledge

But when it comes to experts:

- « we only want an expert with only one hand »
- « One the one hand, the viration is large, so the bearing should be in states $\{S_4, S_5\}$; but, on the other hand, visually inspection of bearing shows no severe wear, so it might be in state S_0.

How to handle the inprecise and partial data from experts? Use evidence theory.

- Assign probability mass on focal elements, rather than basic events.
- Example: $\{A, B\} \rightarrow \{\emptyset, \{A\}, \{B\}, \{A, B\}\}$.
- Allow combining different evidence.

$$(m_1 \oplus m_2)(A) = \frac{1}{1-\kappa} \sum_{B \cap C = A} m_1(B) m_2(C) \qquad \kappa = \sum_{B \cap C = \emptyset} m_1(B) m_2(C).$$

Literature Review

Existing Works on (Hidden) Markov Model-based Prognostics

- **(Hidden) Markov models**

  Markov models: (Liu et al., 2015); (Liu and Chen., 2017);
  Hidden Markov models: (A. Giantomassi, 2011);
  (Ramasso, Denoeux, 2014); (Chinnam, Baruah, 2009); (Wang 2005)
  (Latora and Marchiori, 2007); (Bunks, et al. 2000); (Zaidi, et al. 2012);
  Gaussian Mixture-HMMs: (Xing et al. 2019); (Tobon-Mejia, et al.2012)

- **Hidden semi-Markov models and Dynamic Bayesian Networks**

  (Dong and He, 2007); (Peng and Dong,2011); (Prytzula and Choi, 2008)

Existing works cannot handle the partial knowledge under the belief function framework

Existing Works on Prognostics under the belief function framework

- **Evidence theory and Degradation model**

  Evidential Hidden Markov Model (Ramasso 2014, Ramasso and Denoeux, 2014);
  Evidence theory and Monte Carlo Simulation (He et al. 2009);
  Uncertainty Representation (Baradi et al., 2015);
  Evidential Similarity (Baradi et al., 2019);

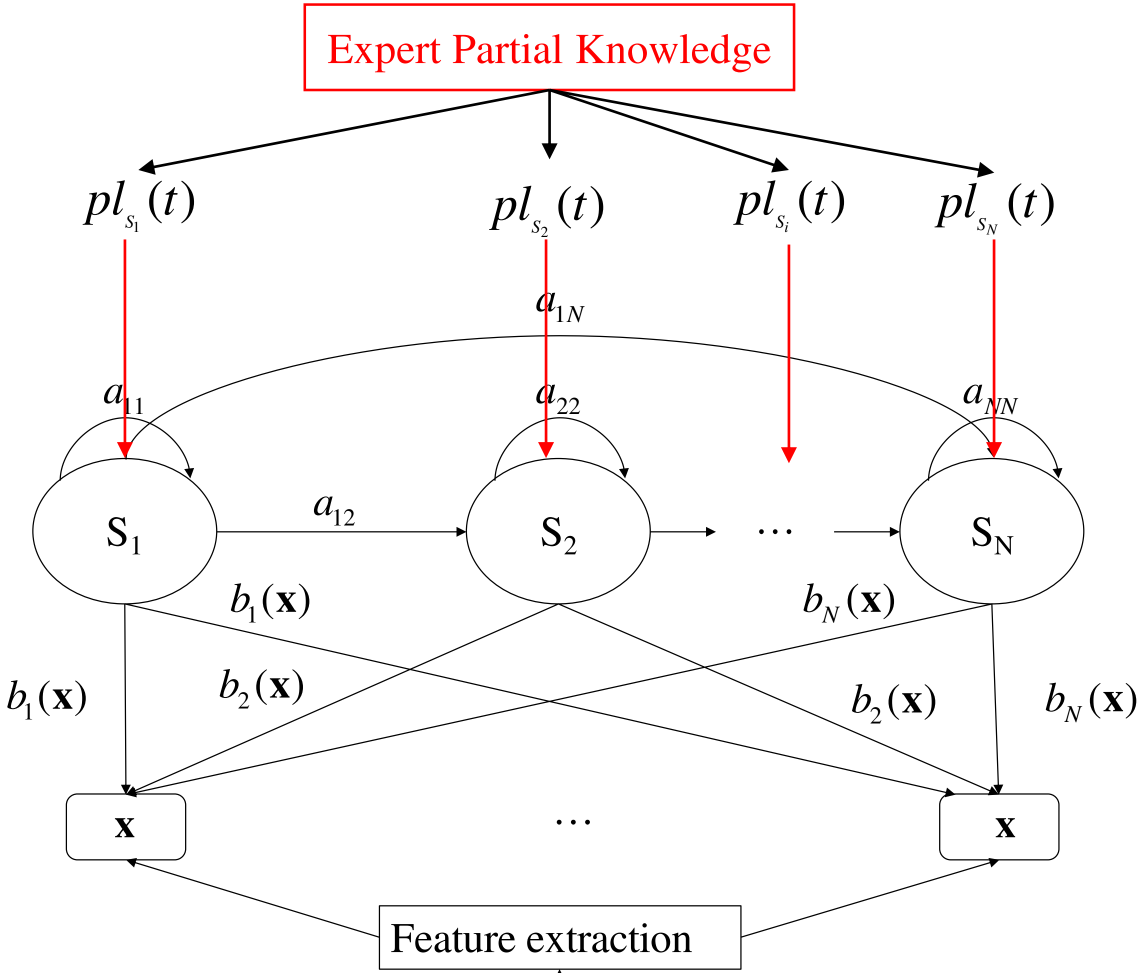
Existing works cannot handle the continuous condition monitoring data and the discrete partial knowledge

Prognostics by Hidden Markov model under the belief function framework haven't been explored to date.

CentraleSupélec

université PARIS-SACLAY

Gaussian Mixture Evidential Hidden Markov Model



Framework of the Gaussian Mixture-Evidential HMMs

Observation Layer: Gaussian Mixture Model

$$b_i(\mathbf{x}) = p(\mathbf{x} \mid S(t) = S_i) = N(\mathbf{x} \mid \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$$

Hidden Layer: Discrete Markov Model

$$p(S(t) = S_i \mid S(t-1), S(t-2), \ldots, S(0)) = p(S(t) = S_i \mid S(t-1))$$

Knowledge Layer: Contour functions (Plausibility on Singletons)

$$(pl(1), pl(2), \ldots, pl(T))$$

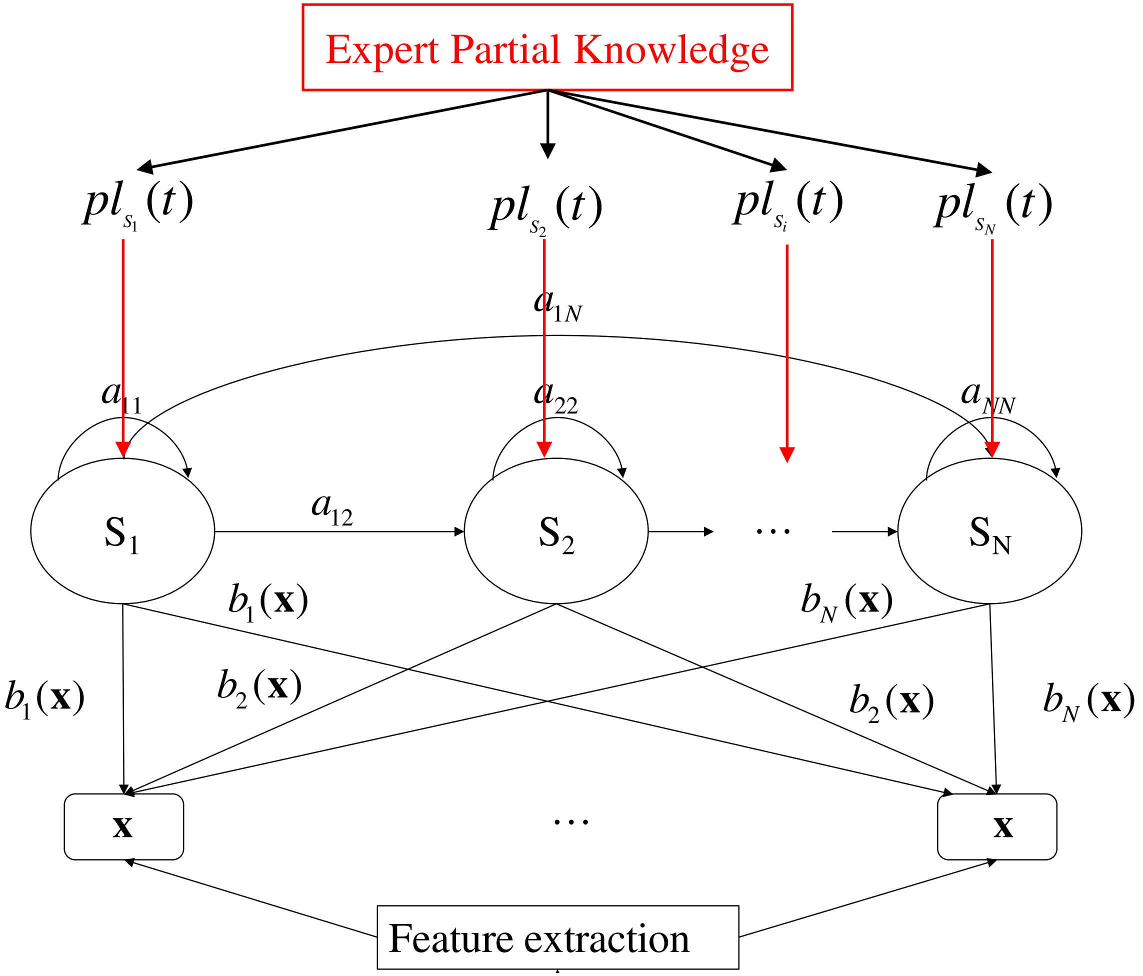
Combination of Probability and Contour function [1]:

$$(p_1 \oplus pl_2)(w) = \frac{p_1(w)\, pl_2(w)}{\sum_{w' \in \Omega} p_1(w)\, pl_2(w)}$$

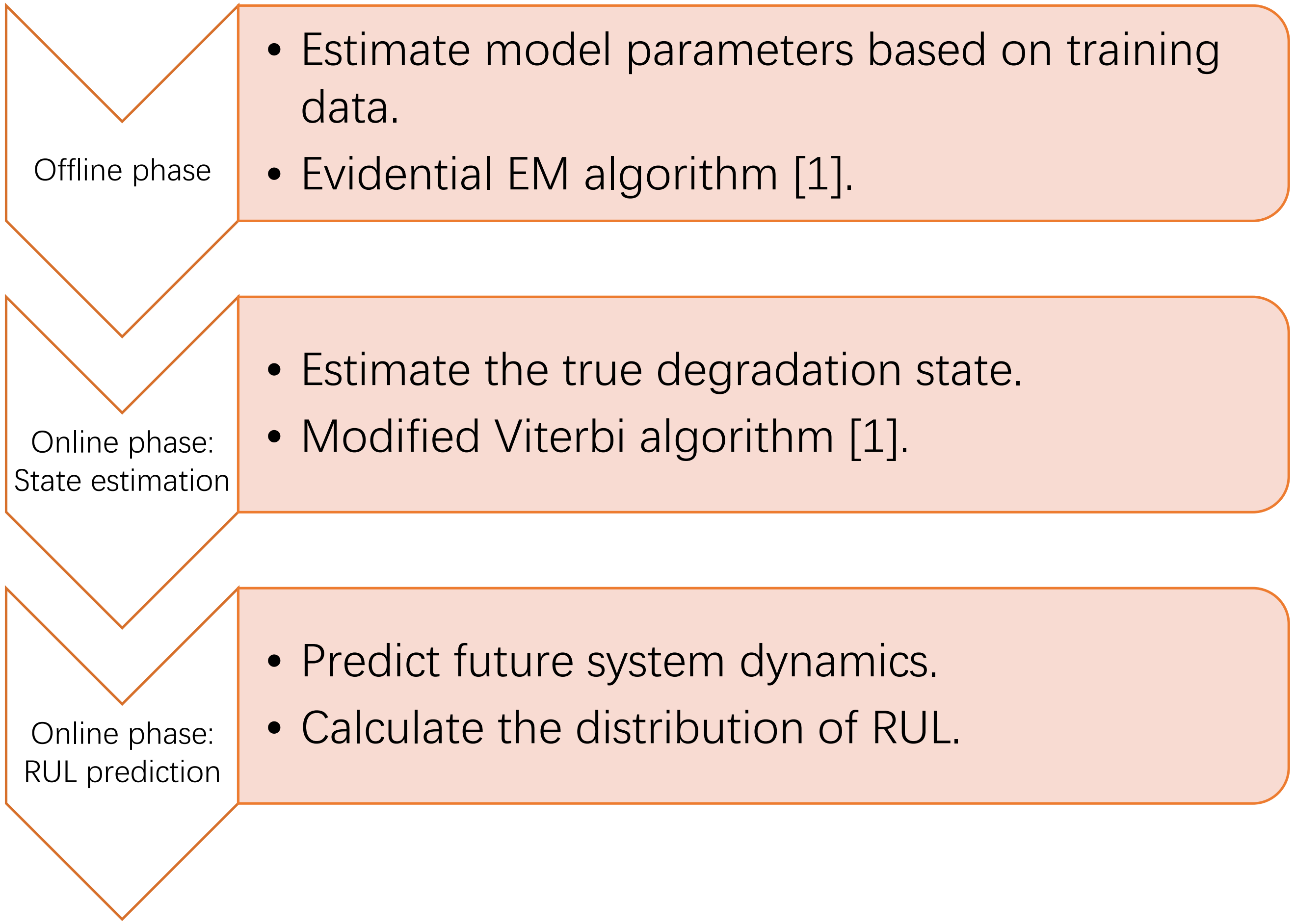
This is also a probability measure

Ramasso, E. and Denoeux, T., 2013. Making use of partial knowledge about hidden states in HMMs: an approach based on belief functions. *IEEE Transactions on Fuzzy Systems*, *22*(2), pp.395-405.

Prognostics by GM-EvHMMs



Framework of the Gaussian Mixture-Evidential HMMs



Procedures of RUL prediction

Ramasso, E. and Denoeux, T., 2013. Making use of partial knowledge about hidden states in HMMs: an approach based on belief functions. *IEEE Transactions on Fuzzy Systems*, *22*(2), pp.395-405.

Prognostics by GM-EvHMMs

Offline phase: Parameter Estimation of the GM-EvHMMs

Solution: The modified Baum-Welch algorithm (Evidential EM algorithm [1])

The modified backward algorithm:

$$\beta_t(S_i) = p(\mathbf{x}(t+1), \mathbf{x}(t+2), ..., \mathbf{x}(T), pl(t+1), pl(t+2), ..., pl(T), S(t) = S_i \mid \pi_i, A, \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

$$= \begin{cases} \beta_T(S_i) = 1 \\ \beta_t(S_i) = [\sum_{j\in\Omega}(b_i(\mathbf{x}(t+1)) \oplus pl(t+1))a_{ji}]\beta_{t+1}(S_i) \end{cases}$$

The inferred parameters:

$$\hat{\pi}_i = \frac{\sum_{k=1}^{n_{Tr}} \gamma_{Tr,t}^{(k)}(S_i)}{n_{Tr}} \qquad \hat{\mu}_i = \frac{\sum_{k=1}^{n_{Tr}} \sum_{t=1}^{t_{Tr}} \gamma_{Tr,t}^{(k)}(S_i)\mathbf{x}_{Tr}^{(k)}(t)}{\sum_{k=1}^{n_{Tr}} \sum_{t=1}^{t_{Tr}} \gamma_{Tr,t}^{(k)}(S_i)} \qquad \hat{\Sigma}_i = \frac{\sum_{k=1}^{n_{Tr}} \sum_{t=1}^{t_{Tr}} \gamma_{Tr,t}^{(k)}(S_i)(\mathbf{x}_{Tr}^{(k)}(t)-\hat{\mu}_i)(\mathbf{x}_{Tr}^{(k)}(t)-\hat{\mu}_i)'}{\sum_{k=1}^{n_{Tr}} \sum_{t=1}^{t_{Tr}} \gamma_{Tr,t}^{(k)}(S_i)}$$

$$\hat{a}_{ij} = \frac{\sum_{k=1}^{n_{Tr}} \sum_{t=1}^{t_{Tr}} \xi_{Tr,t}^{(k)}(S_i, S_j)}{\sum_{k=1}^{n_{Tr}} \sum_{t=1}^{t_{Tr}} \gamma_{Tr,t}^{(k)}(S_i)}$$

$\xi_{Tr,t}^{(k)}(S_i, S_j)$ $\gamma_{Tr,t}^{(k)}(S_i)$ has the same formula with traditional HMMs

Ramasso, E. and Denoeux, T., 2013. Making use of partial knowledge about hidden states in HMMs: an approach based on belief functions. *IEEE Transactions on Fuzzy Systems*, *22*(2), pp.395-405.

Prognostics by GM-EvHMMs

Online phase: Most likely state estimation

Solution: The modified Viterbi algorithm [1]

If condition monitoring data from online can be collected, the modified forward algorithmcan be used:

$$p_{CMD,t}(S_i) = p(S(t) = S_i \mid \mathbf{x}(1), \mathbf{x}(2), ..., \mathbf{x}(t), pl(1), pl(2), ..., pl(t), \hat{\pi}_i, \hat{A}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$$
$$= \frac{\alpha_t(S_i)}{\sum_{i=1}^{N} \alpha_t(S_i)}$$

The most likely state at t given the condition monitoring data up to t can be estimated by:

$$S_{CMD,t} = \underset{i=1,2,...,N}{\arg\max}\ p_{CMD,t}(S_i)$$

← The modified Viterbi algorithm

Ramasso, E. and Denoeux, T., 2013. Making use of partial knowledge about hidden states in HMMs: an approach based on belief functions. *IEEE Transactions on Fuzzy Systems*, *22*(2), pp.395-405.

Prognostics by GM-EvHMMs

Online phase: RUL prediction

Dynamic reliability assessment:

$$R(t_k) = \sum_{S \in W} p_{CMD,t}(S) \times \hat{A}^{(t_k - t)}$$

where *W* is the set of acceptable (working) states

Probability mass function of RUL:

The probability of the system fails in n steps:

$$p_{RUL,predict}(n) = \sum_{S \in W} p_{CMD,t}(S) \times \hat{A}^{(n-t)} - \sum_{S \in W} p_{CMD,t}(S) \times \hat{A}^{(n-1-t)}$$

Metric for the accuracy of RUL prediction (Bhattacharyya Distance):

$$D_B(p_{RUL,true}, p_{RUL,predict}) = -\ln\left(\sum_{t_i=0}^{+\infty} \sqrt{p_{RUL,true}(t_i) \times p_{RUL,predict}(t_i)}\right)$$

Where *B distance* is the similarity of two probability distribution.

Numerical Example——GM-EvHMM Training

Using the following GMM-HMM to generate 100 training sample

$$\mathbf{A} = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 \\ 0 & 0.6354 & 0.3646 & 0 \\ 0 & 0 & 0.7565 & 0.2435 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \boldsymbol{\mu} = \begin{bmatrix} 0.0412 & 0.0916 & 0.0579 \\ 0.1176 & 0.1184 & 0.9168 \\ 0.2002 & 0.2634 & 0.8672 \\ 1 & 1 & 0.8446 \end{bmatrix}$$

$$\Sigma^{(1)} = \begin{bmatrix} 0.0108 & 0.0018 & 0.0007 \\ 0.0018 & 0.0137 & 0.0014 \\ 0.0007 & 0.0014 & 0.0121 \end{bmatrix} \quad \Sigma^{(2)} = \begin{bmatrix} 0.0111 & 0.0020 & 0.0012 \\ 0.0020 & 0.0134 & 0.0019 \\ 0.0012 & 0.0019 & 0.0137 \end{bmatrix}$$

$$\Sigma^{(3)} = \begin{bmatrix} 0.0129 & 0.0039 & 0.0002 \\ 0.0039 & 0.0153 & 0.0002 \\ 0.0002 & 0.0002 & 0.0106 \end{bmatrix} \quad \Sigma^{(4)} = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}$$

True values of the parameters

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.4819 & 0.5181 & 0 & 0 \\ 0 & 0.6364 & 0.3636 & 0 \\ 0 & 0 & 0.7642 & 0.2358 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \hat{\boldsymbol{\mu}} = \begin{bmatrix} 0.0575 & 0.1056 & 0.0703 \\ 0.1249 & 0.1247 & 0.9032 \\ 0.2007 & 0.2861 & 0.8672 \\ 1.0003 & 0.9984 & 0.8448 \end{bmatrix}$$

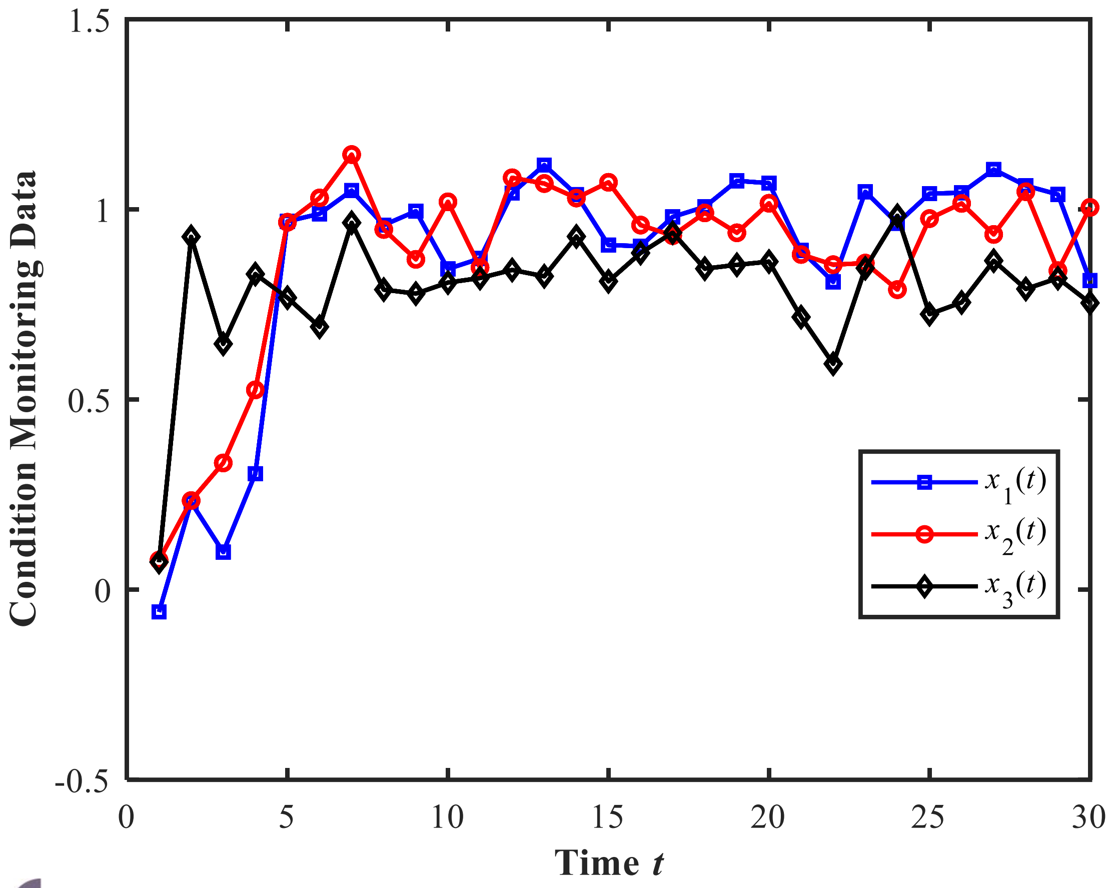
$$\hat{\Sigma}^{(1)} = \begin{bmatrix} 0.0117 & 0.0040 & -0.0025 \\ 0.0040 & 0.0157 & 0.0021 \\ -0.0025 & 0.0021 & 0.0143 \end{bmatrix} \quad \hat{\Sigma}^{(2)} = \begin{bmatrix} 0.0118 & 0.0041 & 0.0022 \\ 0.0041 & 0.0151 & 0.0010 \\ 0.0022 & 0.0010 & 0.0129 \end{bmatrix}$$

$$\hat{\Sigma}^{(3)} = \begin{bmatrix} 0.0108 & 0.0025 & -0.0003 \\ 0.0025 & 0.0157 & 0.0016 \\ -0.0003 & 0.0016 & 0.0116 \end{bmatrix} \quad \Sigma^{(4)} = \begin{bmatrix} 0.0105 & 0 & 0 \\ 0 & 0.0103 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}$$

Estimated values of the parameters (No expert judgement data)

Numerical Example——GM-EvHMM Testing

Using the True parameters to generate Condition Monitoring Data (CMD)



Time	t1	t2	t3	t4	t5
True state	1	1	2	2	2
State by CMD	1	1	2	3	3
State by precise knowledge	1	1	2	2	2

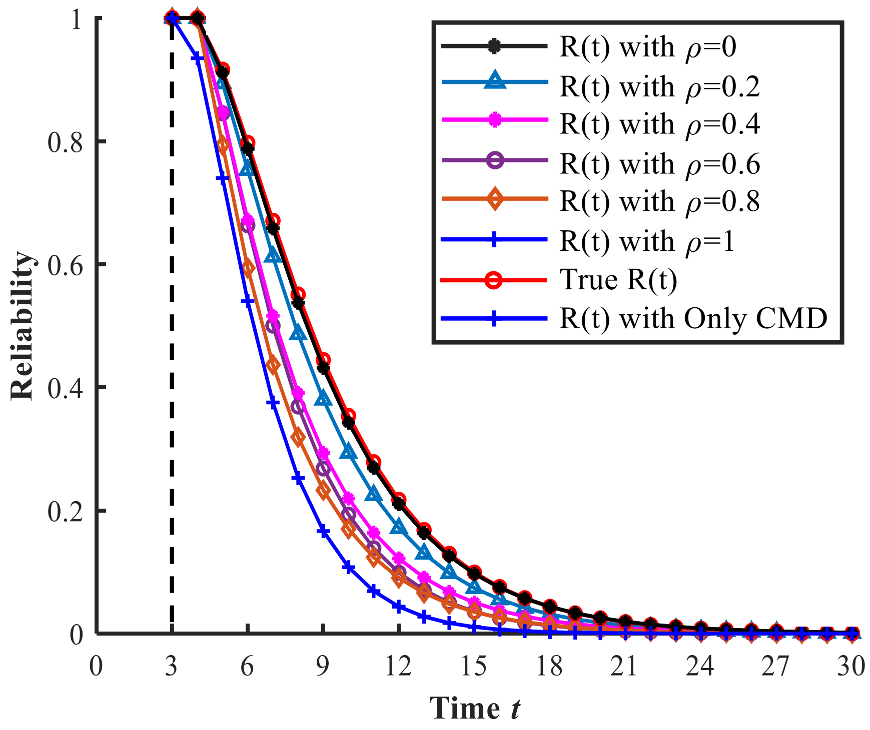
Time	t6	t7	t8	t9	t10-t30
True state	3	3	3	4	4
State by CMD	3	3	3	4	4
State by precise knowledge	3	3	3	4	4

Adding Partial Knowledge in Offline Phase

The Partial Knowledge has the following form:

$$pl(t) = \begin{cases} 1 & \text{if } y_t = S_i \\ \rho & \text{otherwise} \end{cases}$$

The true state has the maximum plausibility, ρ is the nonspecificity coefficient, quantifies the uncertainty of the true label (state).



Results: With the nonspecificity approaches to 1, which means the most uncertainty (No knowledge), the reliability results can be more imprecise, but still better than without any knowledge about the hidden states.

Adding Partial Knowledge with Noise in Offline Phase

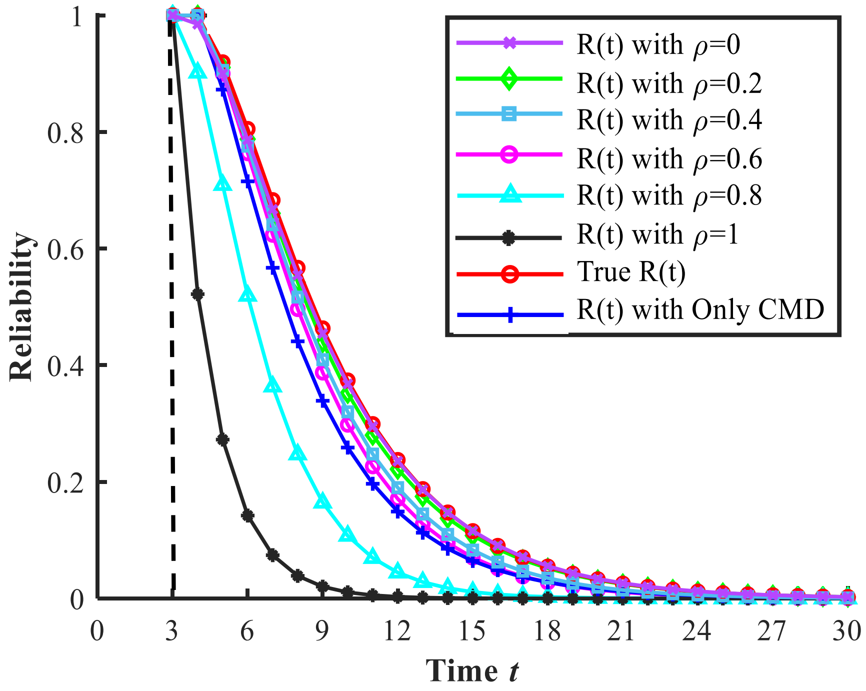
The Partial Knowledge with noise has the following form:

$$Error\ probability \sim Beta(\rho, 0.2)$$

$$pl(t) = \begin{cases} 1 & \text{Random selected from } S_i \text{ where } S_i \neq (y_t = S_j) \\ \rho & \text{Otherwise} \end{cases}$$

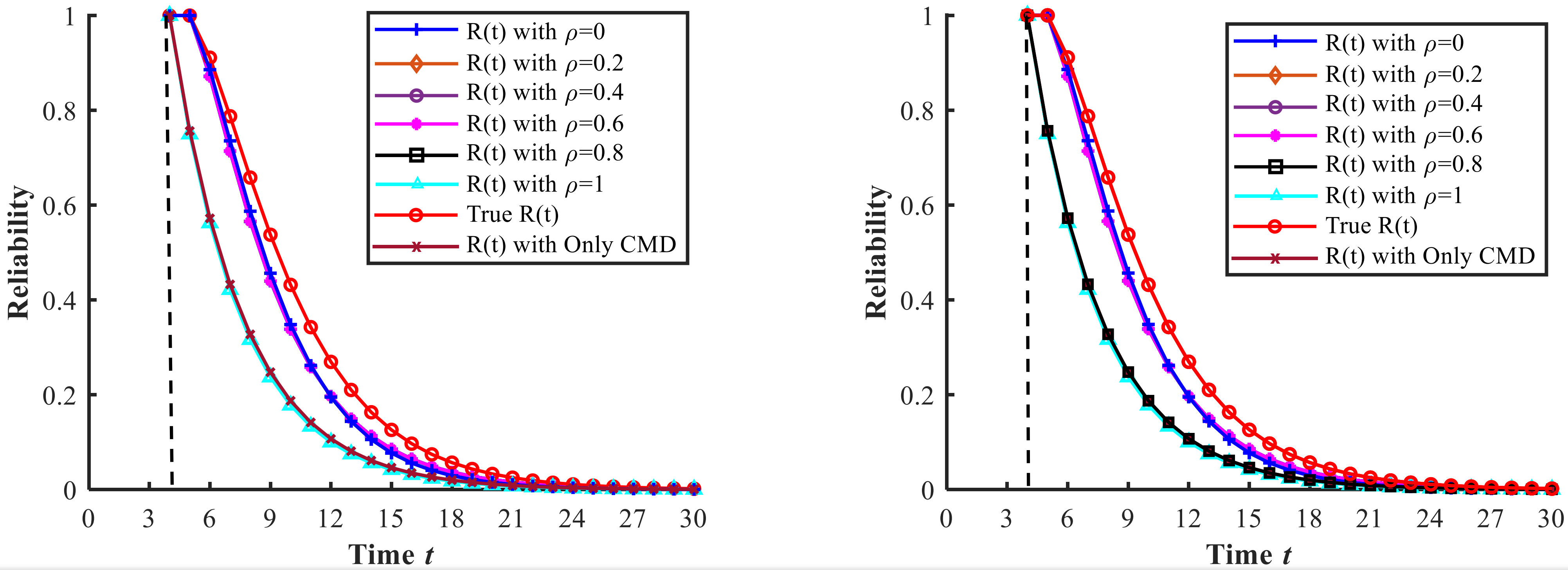
The state with the maximum plausibility by experts is not the true state , quantifies the noise of the true label (state).

Results: Even an error probability with 0.6, the reliability is more accurate than without knowledge, when the error proability approaches to 1, the predicted state is wrong with a high probability.



Adding Partial Knowledge and Noise Label in Online Phase

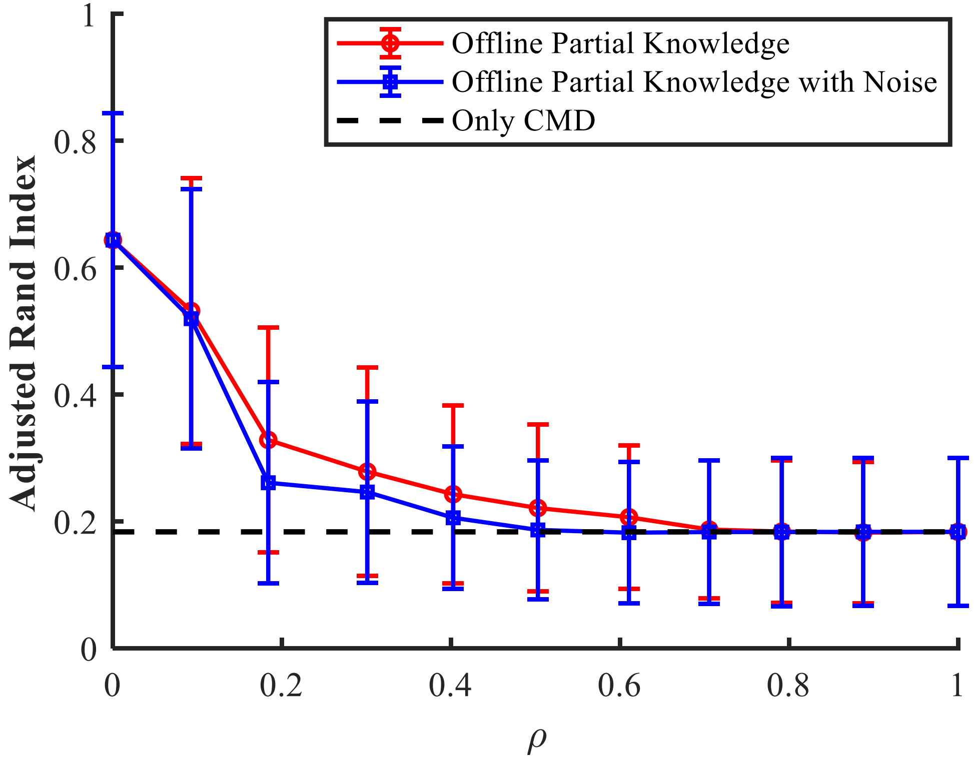
The estimated state at t=4 is wrong only by CMD:

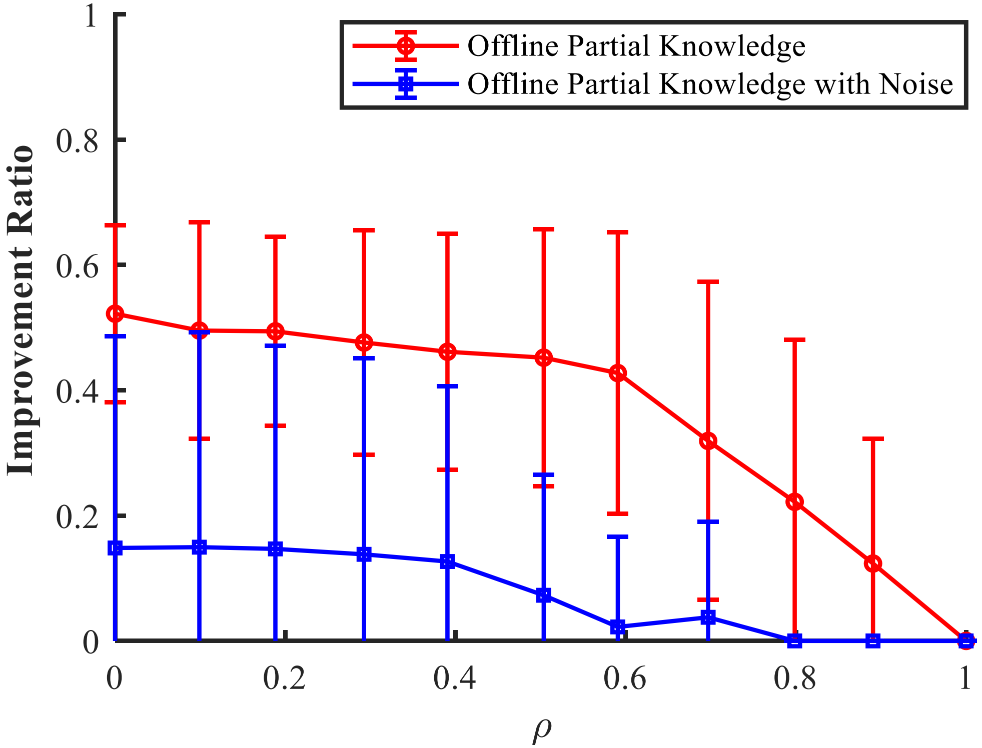


Remarks: In online phase, the partial knowledge can improve the accuracy of state estimation with CMD. Also, partial knowledge with noise with 0.6 error probability , the accuracy of state estimation can also be improved.

Adding Partial Knowledge in Offline Phase

Examine the state estimation performance





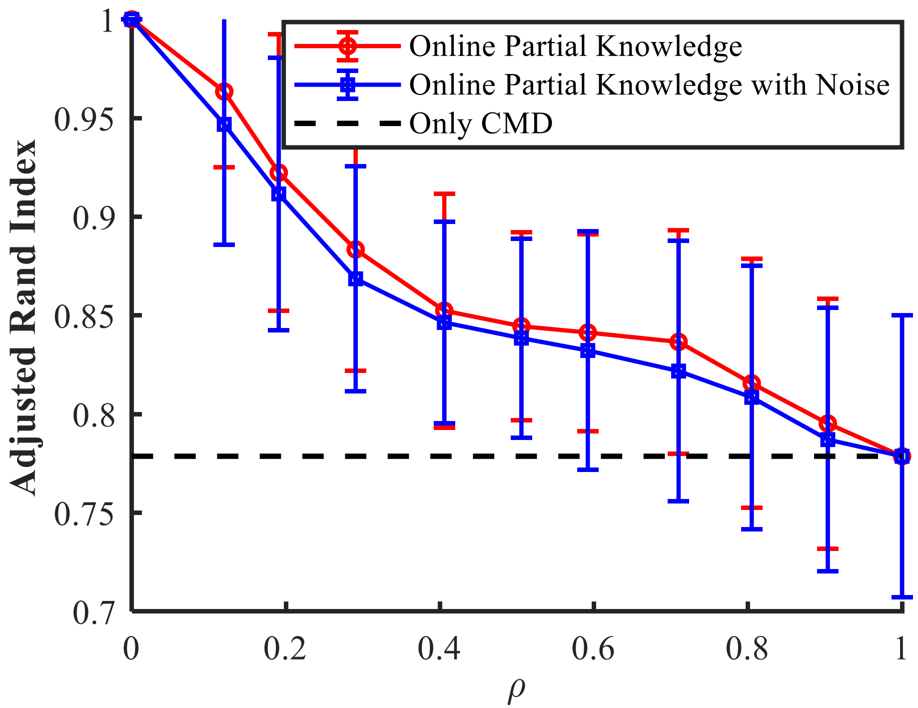
$$ARI = \frac{RI - E(RI)}{\max(RI) - E(RI)}$$

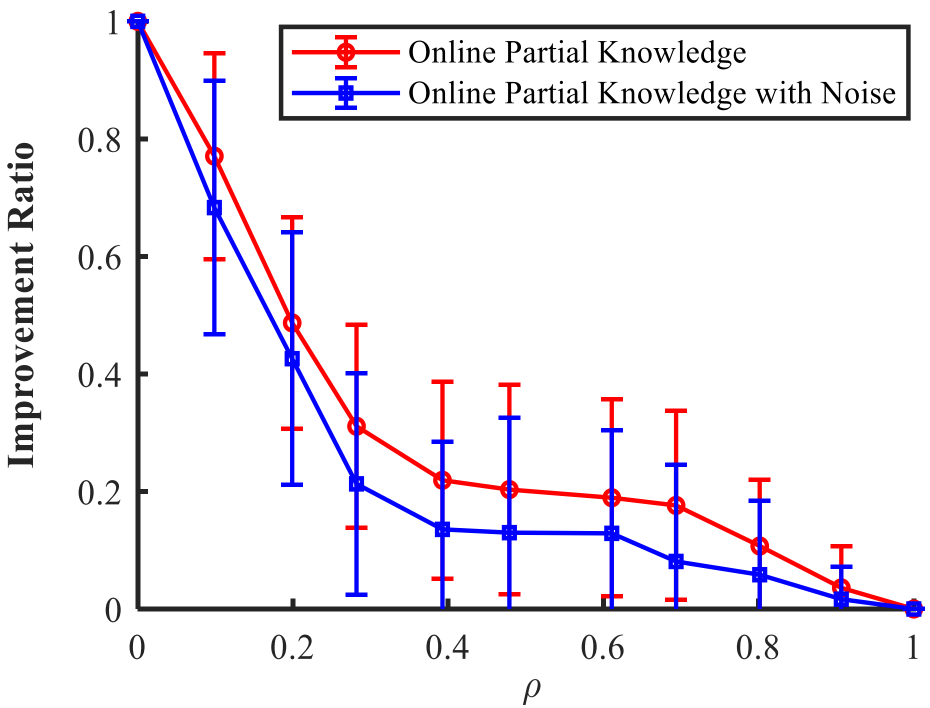
$$IR = \frac{Error\ number\ only\ with\ CMD - Error\ number\ with\ partial\ knowledge}{Error\ number\ only\ with\ CMD}$$

RI (Rand Index) is the correct classification ratio

Adding Partial Knowledge with Noise in Online Phase

Examine the state estimation performance

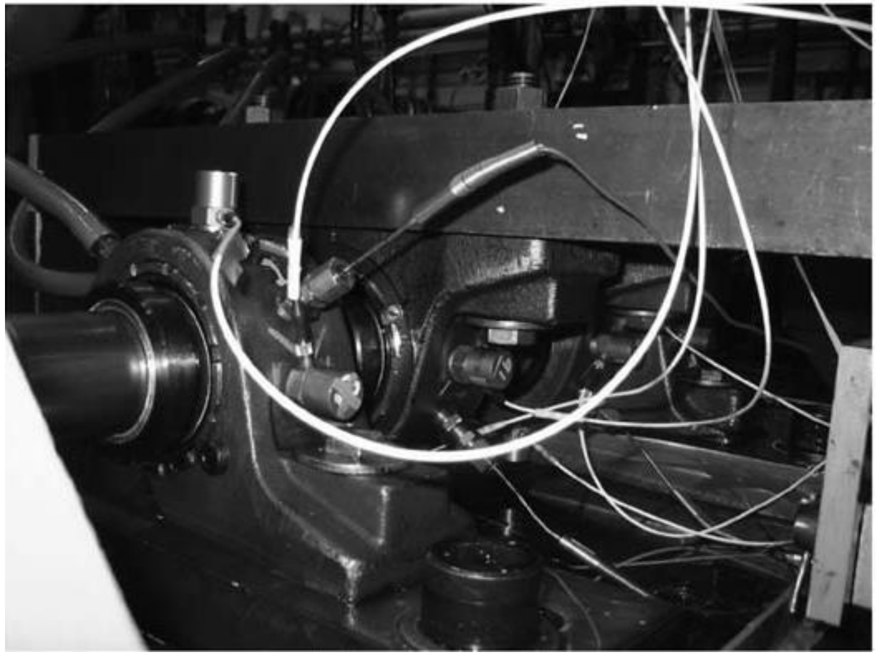


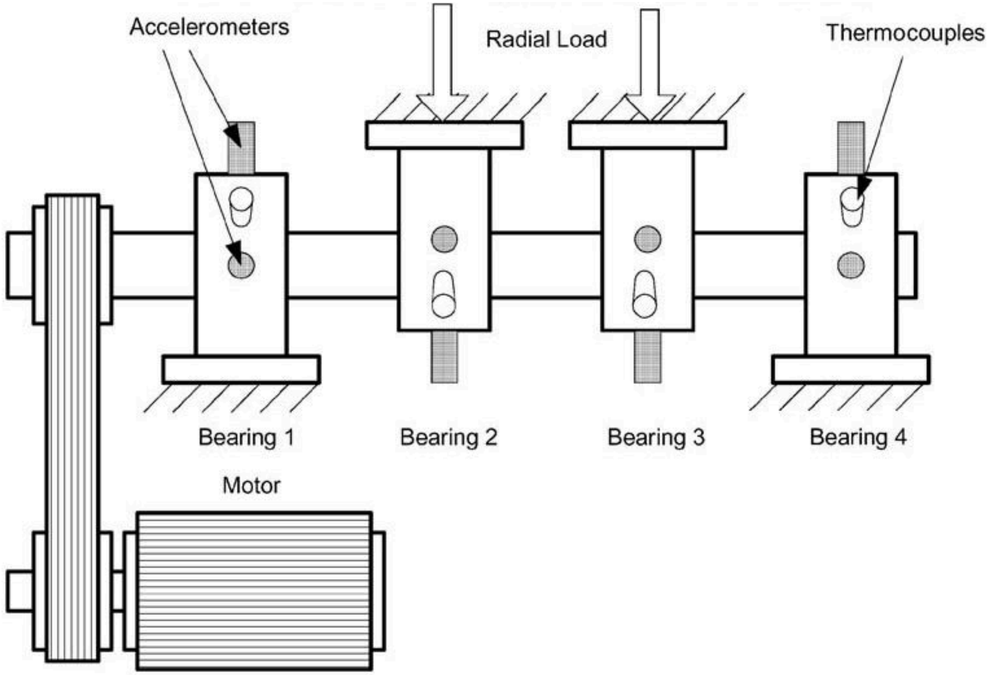


Remarks: In online phase, the partial knowledge (with noise) can improve the accuracy of state estimation with CMD. Also the accuracy of state estimation by partial knowledge is larger than the state estimation by partial knowledge with noise.

Application to Bearing Test Data by NASA

- Four bearings with each has 2 vibration accelerometers (one vertical, one horizontal);
- The viabration data is the raw signals with every 5 minites;
- Only the Bearing 3 has ran to failure.





The bearing test rig

CentraleSupélec

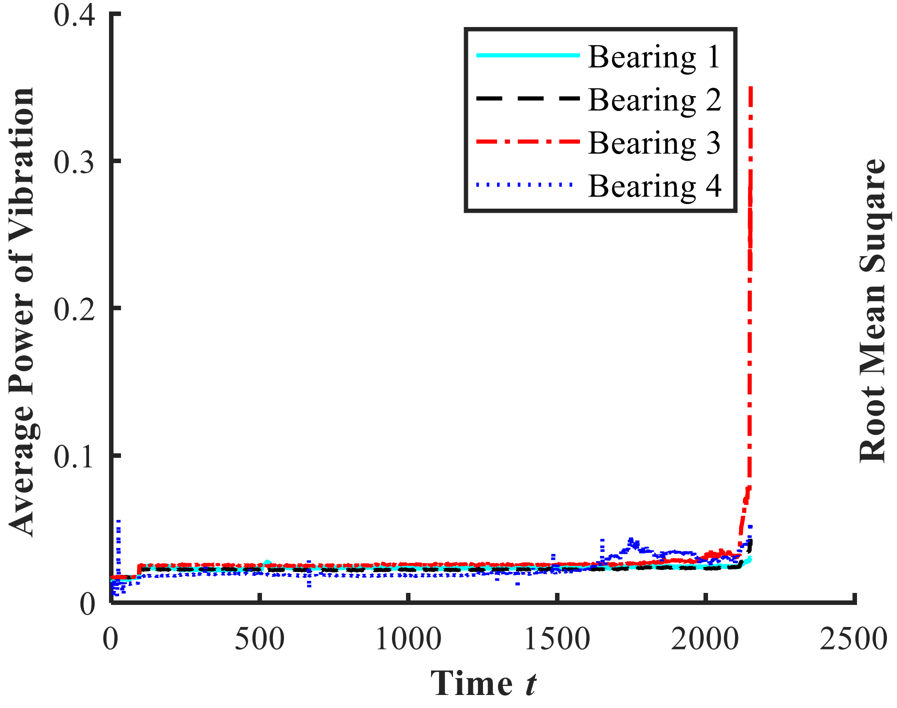
universite PARIS-SACLAY

Application to Bearing Test Data by NASA

- **Three features** are selected :

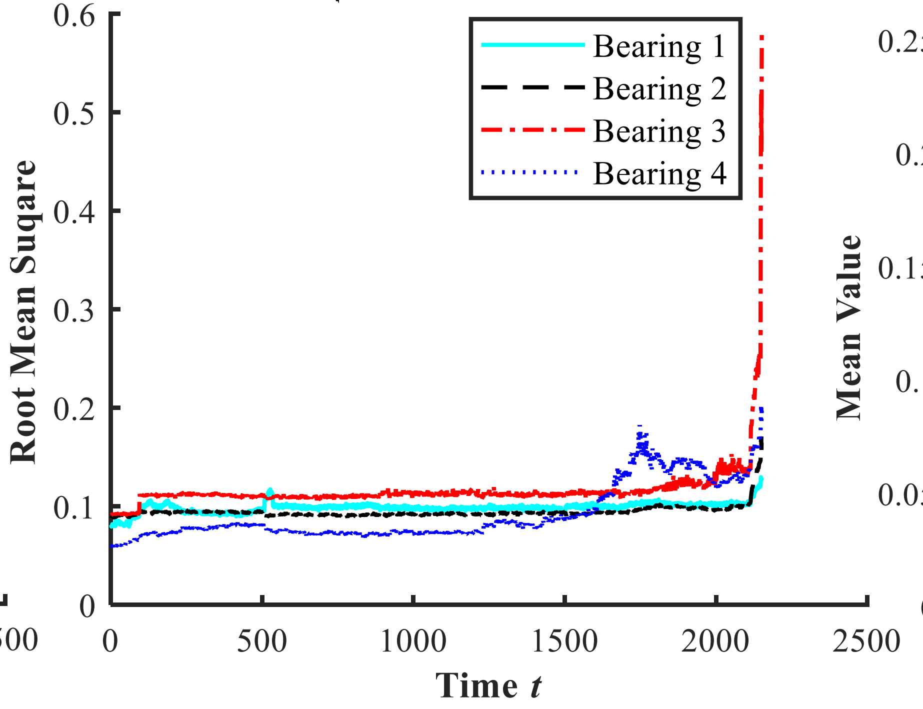
Average Power of Vibration

$$x_1(t_i) = \frac{1}{(t_i - t_{i-1})f} \sum_{j \in (t_{i-1}, t_i)} c_j^2$$



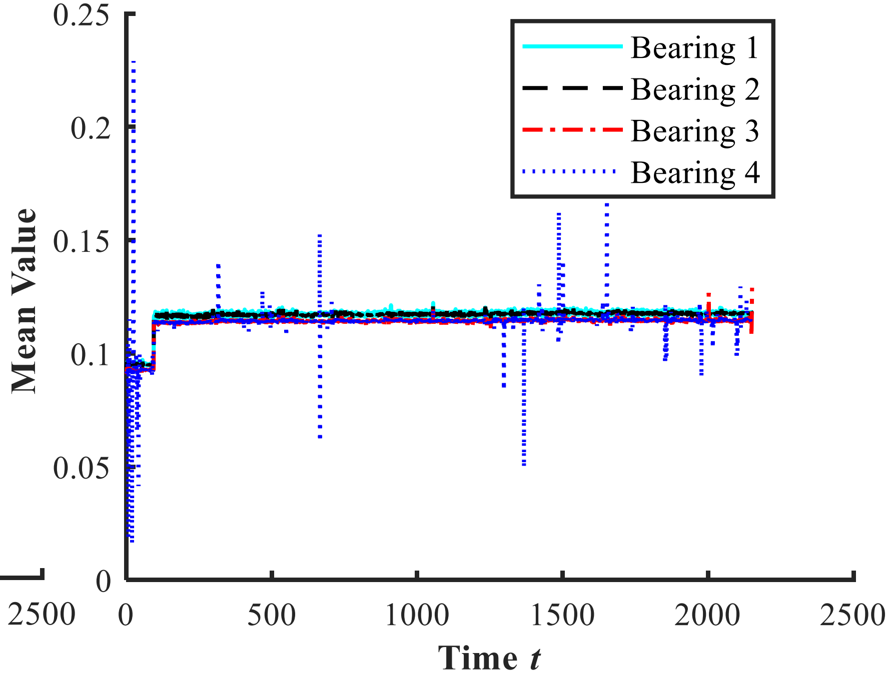
Root Mean Square Error

$$x_2(t_i) = \sqrt{\frac{1}{(t_i - t_{i-1})f} \sum_{j \in (t_{i-1}, t_i)} (c_j - \bar{c})^2}$$



Mean Value

$$x_3(t_i) = \frac{1}{(t_i - t_{i-1})f} \sum_{j \in (t_{i-1}, t_i)} c_j$$

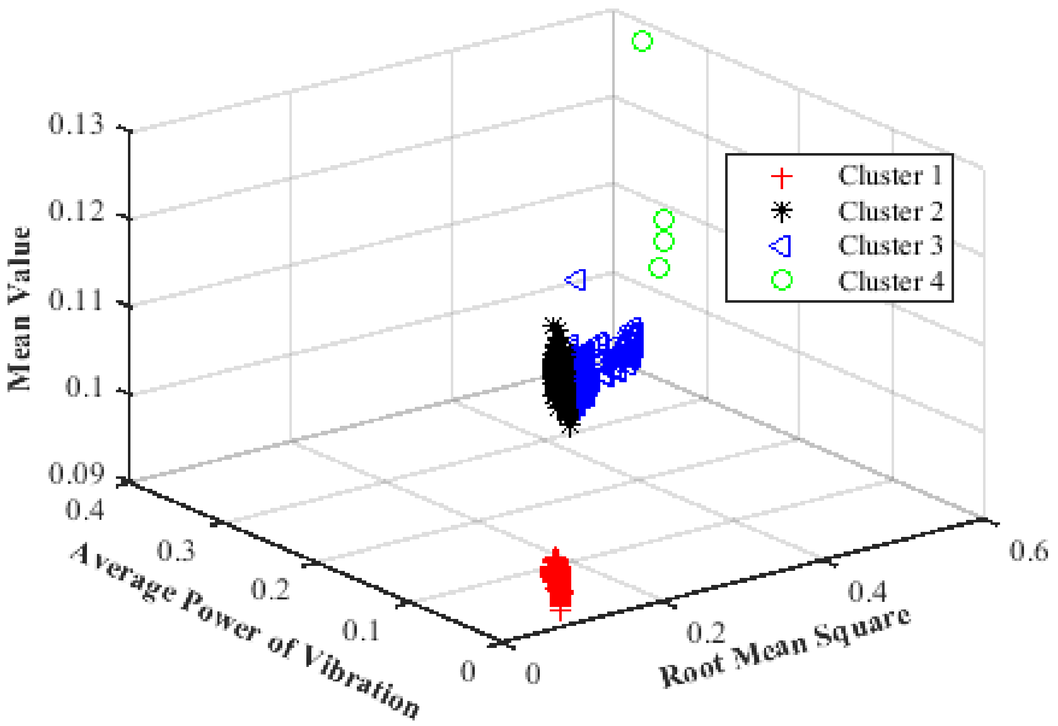


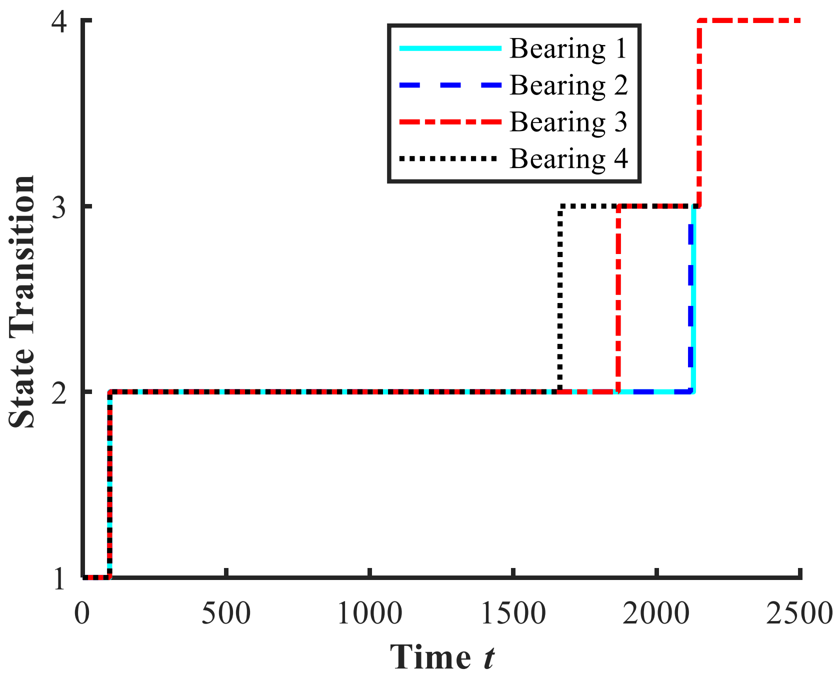
CentraleSupélec

université PARIS-SACLAY

Clustering by Gaussian Mixture Model

State Clustering by GMM and its clustering results.





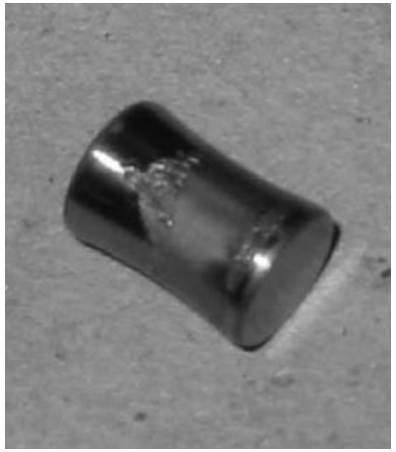
CentraleSupélec

université PARIS-SACLAY

State Division and Verification by GMM Clustering

State 1: Perfect Working

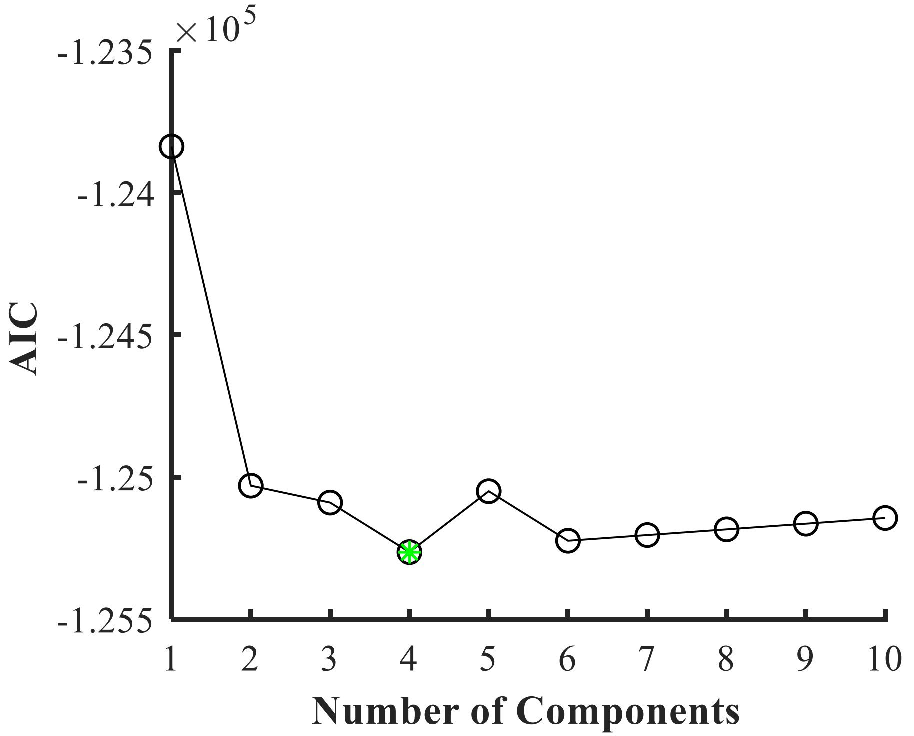
State 2: Minor degradation



State 3: Medium degradation

State 4: Severe degradation

State division by human



CentraleSupélec

université PARIS-SACLAY

Estimate GMM-EvHMM by 100 Bootstrap samples

$$\mathbf{A}=\begin{bmatrix}0.9889 & 0.0111 & 0 & 0\\ 0 & 0.9998 & 0.0002 & 0\\ 0 & 0 & 0.9688 & 0.0312\\ 0 & 0 & 0 & 1\end{bmatrix} \quad \boldsymbol{\mu}=\begin{bmatrix}0.0177 & 0.0975 & 0.0904\\ 0.0242 & 0.1163 & 0.1028\\ 0.0640 & 0.1148 & 0.2237\\ 0.2759 & 0.1167 & 0.5098\end{bmatrix}$$

$$\Sigma^{(1)}=10^{-4}\times\begin{bmatrix}0.0586 & -0.1699 & 0.1205\\ -0.1699 & 0.5802 & -0.2499\\ 0.1205 & -0.2499 & 0.3624\end{bmatrix} \quad \Sigma^{(2)}=10^{-3}\times\begin{bmatrix}0.0048 & 0.0004 & 0.0221\\ 0.0004 & 0.0068 & 0.0086\\ 0.0221 & 0.0086 & 0.1096\end{bmatrix}$$

$$\Sigma^{(3)}=10^{-3}\times\begin{bmatrix}0.1448 & -0.0015 & 0.3255\\ -0.0015 & 0.0008 & -0.0032\\ 0.3255 & -0.0032 & 0.7506\end{bmatrix} \quad \Sigma^{(4)}=\begin{bmatrix}0.0025 & 0.0047 & 0.0024\\ 0.0047 & 0.0110 & 0.0043\\ 0.0024 & 0.0043 & 0.0023\end{bmatrix}$$

(Parameter estimation by the four bearing data)

$$\hat{\mathbf{A}}=\begin{bmatrix}0.9902 & 0.0098 & 0 & 0\\ 0 & 0.9998 & 0.0002 & 0\\ 0 & 0 & 0.9959 & 0.0041\\ 0 & 0 & 0 & 1\end{bmatrix} \quad \hat{\boldsymbol{\mu}}=\begin{bmatrix}0.0170 & 0.0977 & 0.0845\\ 0.0240 & 0.1158 & 0.1010\\ 0.1250 & 0.1174 & 0.3472\\ 0.1933 & 0.1186 & 0.3206\end{bmatrix}$$

$$\hat{\Sigma}^{(1)}=10^{-3}\times\begin{bmatrix}0.1644 & -0.0003 & -0.0019\\ -0.0003 & 0.0066 & -0.0008\\ -0.0019 & -0.0008 & 0.1331\end{bmatrix} \quad \hat{\Sigma}^{(2)}=10^{-3}\times\begin{bmatrix}0.3181 & 0.0313 & -0.0163\\ 0.0313 & 0.0252 & -0.0070\\ -0.0163 & -0.0070 & 0.0324\end{bmatrix}$$

$$\hat{\Sigma}^{(3)}=\begin{bmatrix}0.0034 & 0.0004 & 0.0000\\ 0.0004 & 0.0013 & 0.0001\\ 0.0000 & 0.0001 & 0.0002\end{bmatrix} \quad \Sigma^{(4)}=\begin{bmatrix}0.0258 & -0.0015 & -0.0017\\ -0.0015 & 0.0152 & 0.0004\\ -0.0017 & 0.0004 & 0.0040\end{bmatrix}$$

(Parameter estimation by Bootstrap sample: True values)

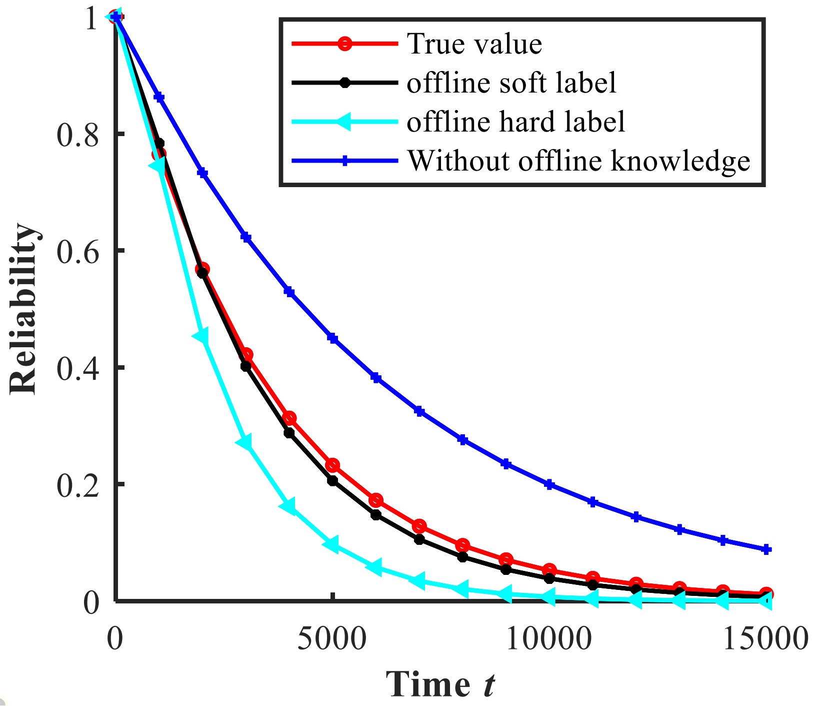
CentraleSupélec

université PARIS-SACLAY

Adding Partical Knowledge in Offline Phase

Offline partial knowledge:

1. Hard labels by the clustering results;
2. Soft labels by the posterior probability of GMM model

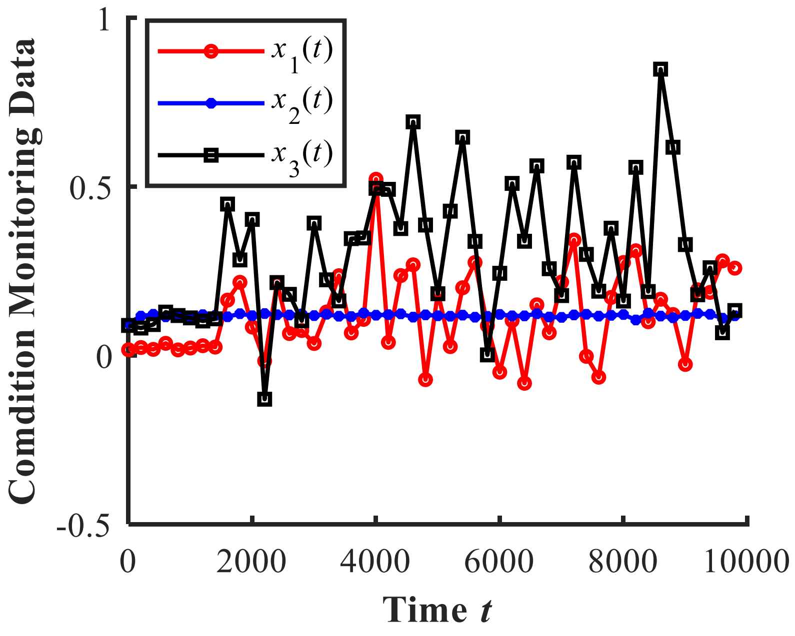


Results:

1. Using offline knowledge, the reliability results are conservative.
2. The results by using soft labels is better than hard labels.

Online Condition Monitoring Data and the State Estimation

Online Condition Monitoring Data (CMD):



State Estimation by CMD:

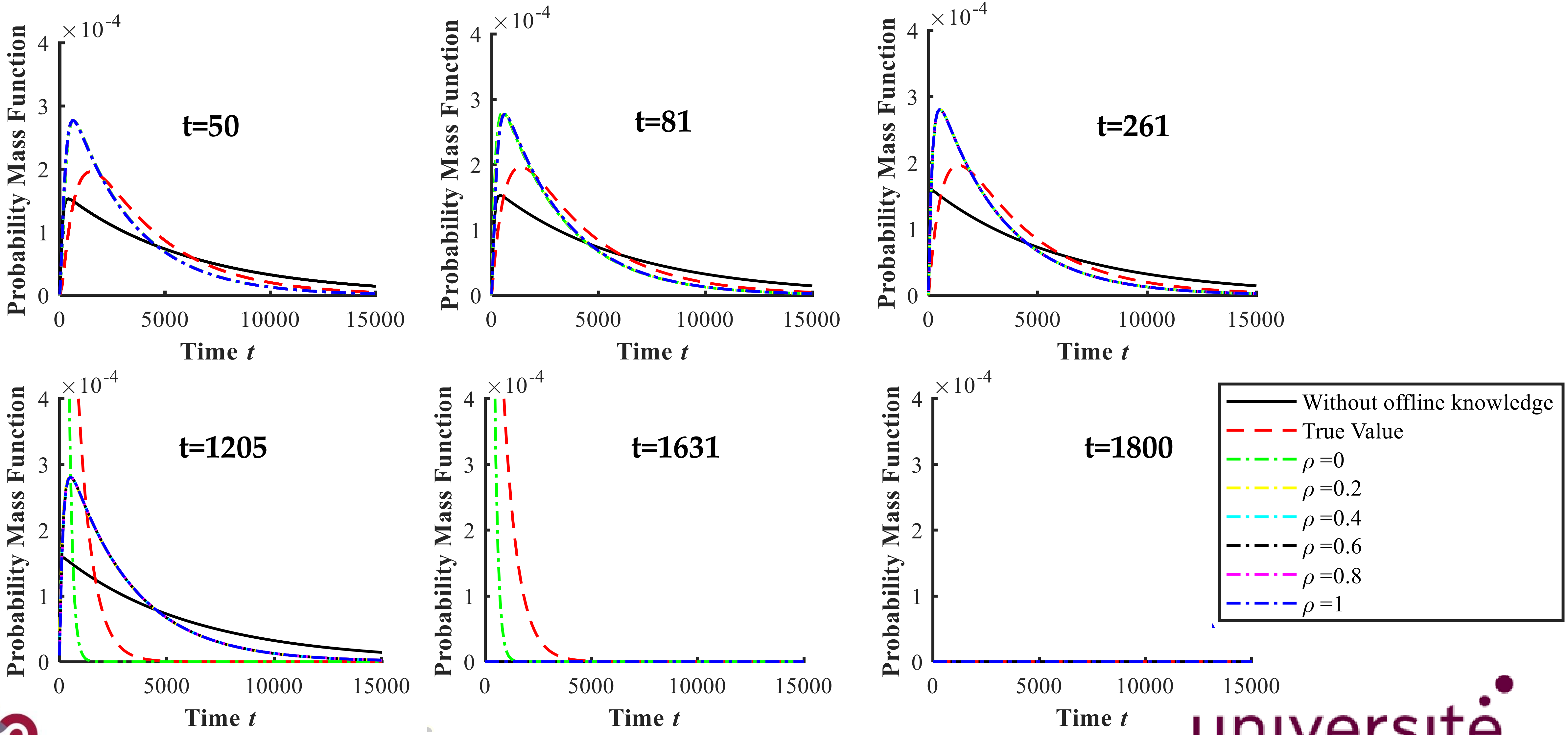
Time	t=50	t=81	t=261	t=1205
True state	1	2	2	3
State by CMD	1	1	2	2
State by precise knowledge	1	2	2	3

Time	t=1631	t=1800	t=1801~t=10000
True state	3	4	4
State by CMD	4	4	4
State by precise knowledge	3	4	4

CentraleSupélec

université PARIS-SACLAY

Adding Precise/Partial Knowledge in Online phase



Probability mass functions of the RUL

CentraleSupélec

université PARIS-SACLAY

Adding Precise/Partial Knowledge in Online phase

$$D_B(p_{RUL,true}, p_{RUL,predict}) = -\ln\left(\sum_{t_i=0}^{+\infty}\sqrt{p_{RUL,true}(t_i)\times p_{RUL,predict}(t_i)}\right)$$

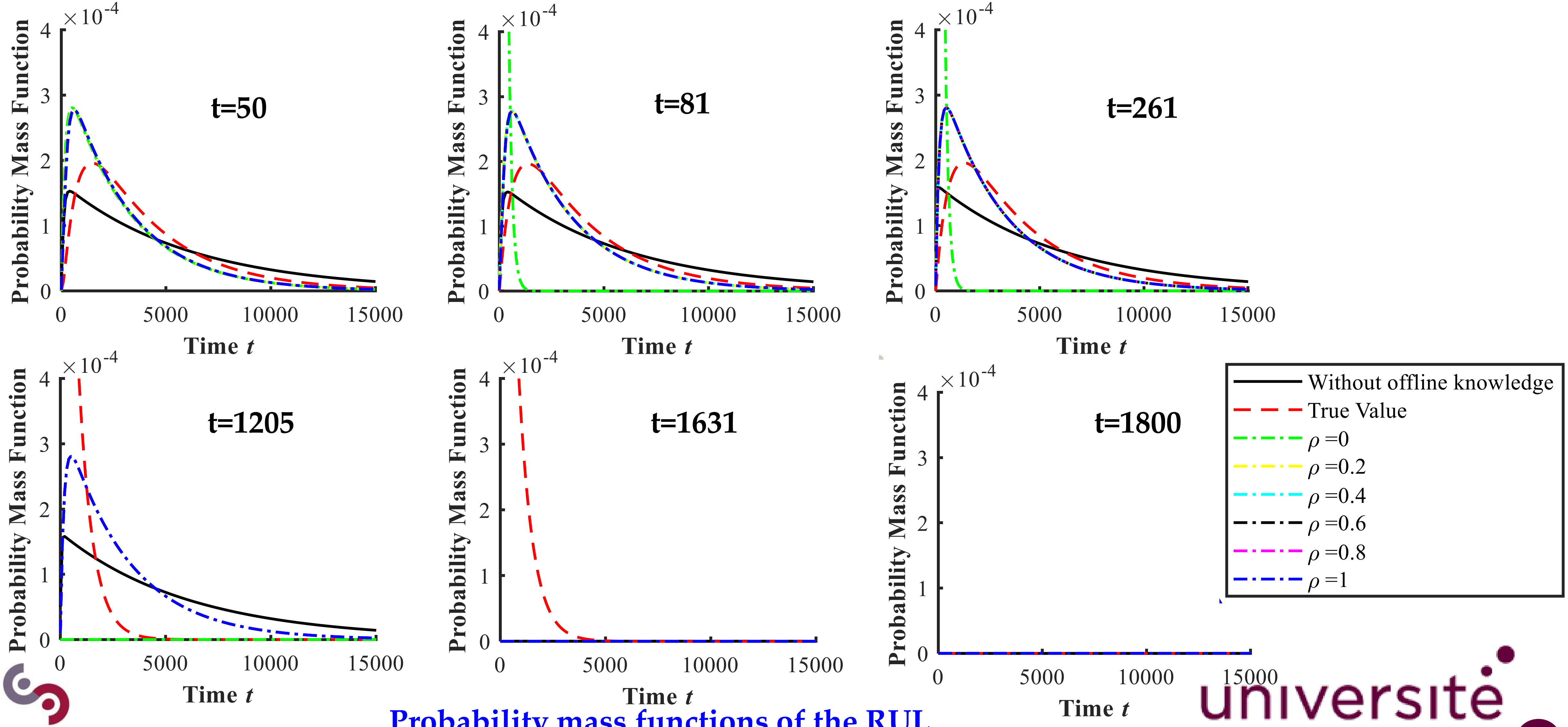
Time *t*	Bhattacharyya Distance						
	$\rho=0$	$\rho=0.2$	$\rho=0.4$	$\rho=0.6$	$\rho=0.8$	$\rho=1$	Without offline knowledge
t=50	**0.017485**	0.017945	0.018360	0.018736	0.019077	0.019387	0.038831
t=81	**0.013333**	0.013471	0.013683	0.014049	0.014811	0.017234	0.035810
t=261	**0.017240**	0.017300	0.017359	0.017419	0.017479	0.017540	0.038328
t=1205	**0.221881**	0.305304	0.311330	0.3135439	0.314705	0.315424	0.523747
t=1631	**0.221881**	+∞	+∞	+∞	+∞	+∞	+∞
t=1800	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Remarks: In online phase, the partial knowledge can improve the accuracy of state estimation with CMD. Also, when the state estimation by CMD is true, the B distance is reduced when time goes on.

CentraleSupélec

université PARIS-SACLAY

Adding Precise/Partial Knowledge with Noise in Online Phase



Probability mass functions of the RUL

Adding Precise/Partial Knowledge with Noise in Online Phase

$$D_B(p_{RUL,true}, p_{RUL,predict}) = -\ln\left(\sum_{t_i=0}^{+\infty}\sqrt{p_{RUL,true}(t_i)\times p_{RUL,predict}(t_i)}\right)$$

Time *t*	**Bhattacharyya Distance**						
	$\rho=0$	$\rho=0.2$	$\rho=0.4$	$\rho=0.6$	$\rho=0.8$	$\rho=1$	Without offline knowledge
t=50	0.025276	0.022275	0.020944	0.020196	0.019718	**0.019387**	0.038831
t=81	1.476586	0.019046	**0.016023**	0.017649	0.017453	0.017234	0.035810
t=261	1.476586	0.018834	0.018015	0.017749	0.017618	**0.017540**	0.038328
t=1205	+∞	**0.315424**	**0.3154246**	**0.315424**	**0.315424**	**0.315424**	0.523747
t=1631	+∞	+∞	+∞	+∞	+∞	+∞	+∞
t=1800	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Remarks: In online phase, the partial knowledge with noise **sometimes can improve** the accuracy of state estimation with CMD (Because of the Dempster's rule).

CentraleSupélec

université PARIS-SACLAY

Conclusion and Future Works

❖ Conclusion

- Gaussian Mixture Evidential Hidden Markov Model (GMM-EvHMM) is proposed
- Remaining Useful Life Prediction is Conducted by Fusing Partial Knowledge
- Performance of the GM-EvHMM is validated by numerical experiment
- The bearing data from NASA is used to demonstrate the effectiveness of the methods

❖ Future Works

- Application to Multi-component systems
- How to fusing the condition monitoring data from multiple physical levels of a system
- How to fusing other types of information, such as the lifetime data, CMD, and expert knowledge

Thank you for your attention!

Your questions/comments are sincerely welcomed!

zhiguo.zeng@centralesupelec.fr