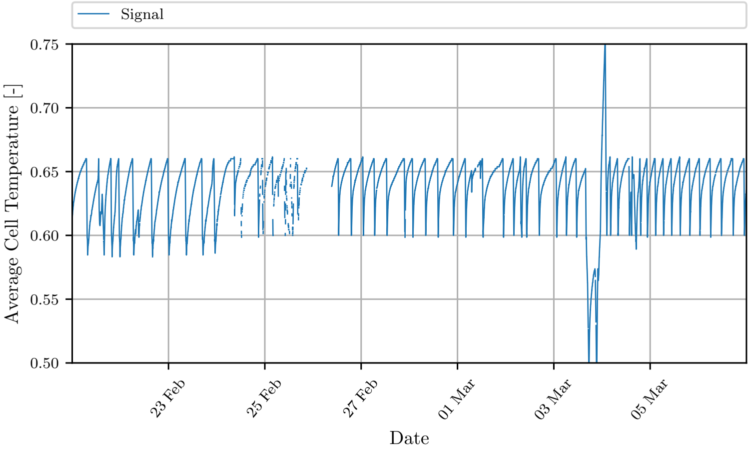
Real-Time Outlier Detection with Dynamic Process Limits

Process Control 2023

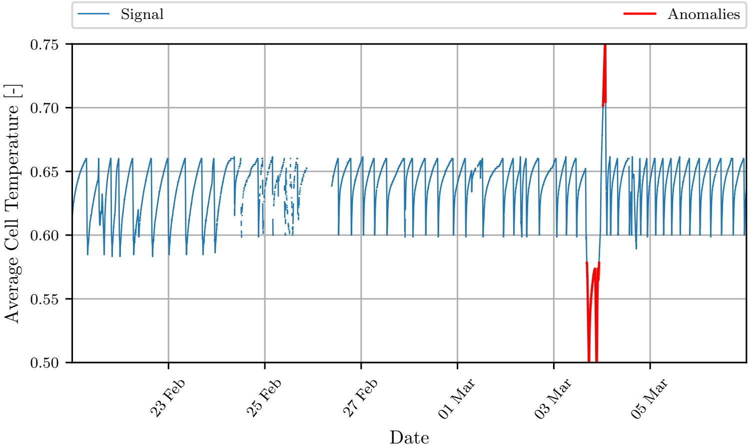
Marek Wadinger, Michal Kvasnica

Institute of Information Engineering, Automation, and Mathematics
Faculty of Chemical and Food Technology
Slovak University of Technology in Bratislava
marek.wadinger@stuba.sk

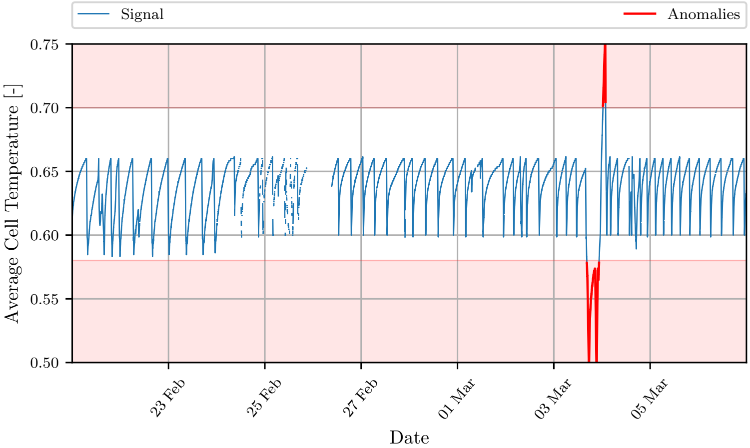
Real World Data



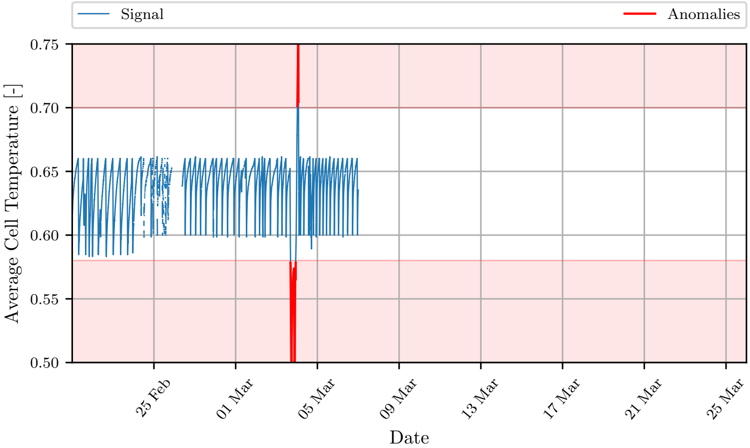
Data with Outliers



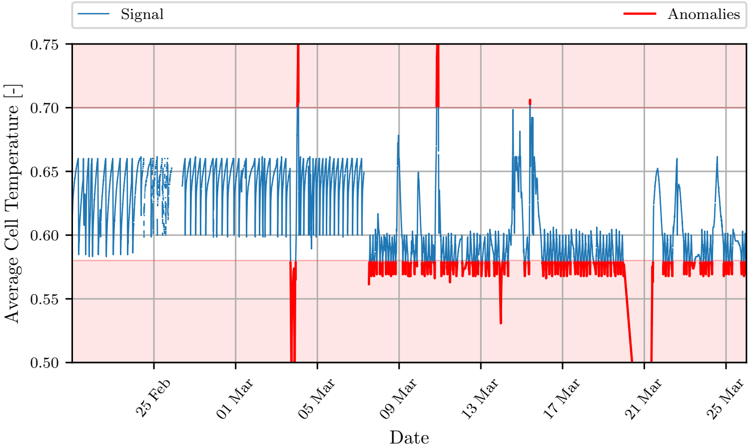
Static Threshold Limits



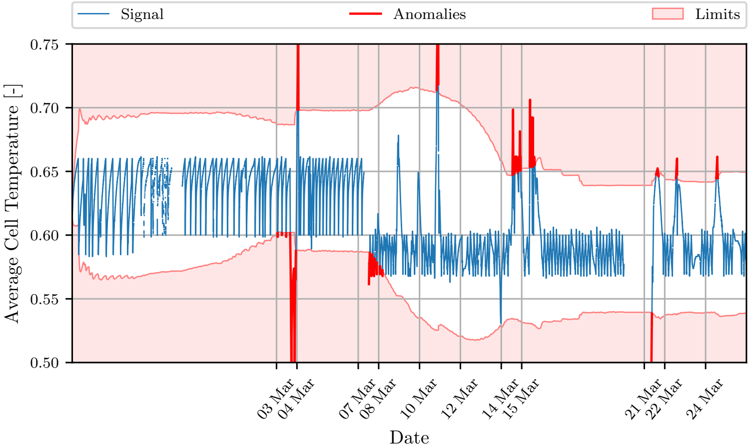
Static Threshold Limits



Comming Problem



Control Engineering Meets Artificial Intelligence



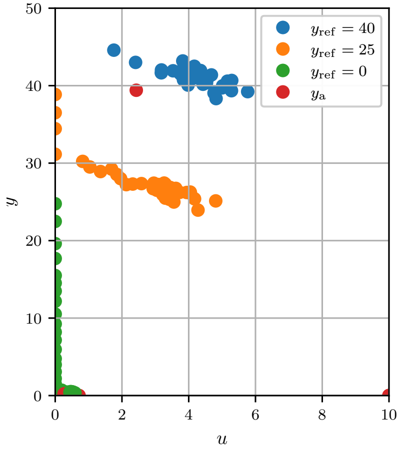
Goals

We need to design a data-driven outlier detector that:

- adapts to unseen operation regime
- provides conservative dynamic process limits
- operates with existing infrastructure

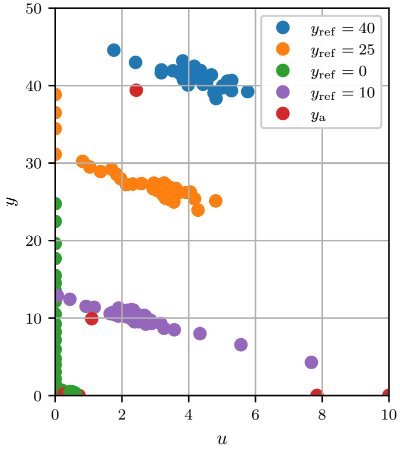
Limitations of Data-driven Outlier Detection

- need for training data



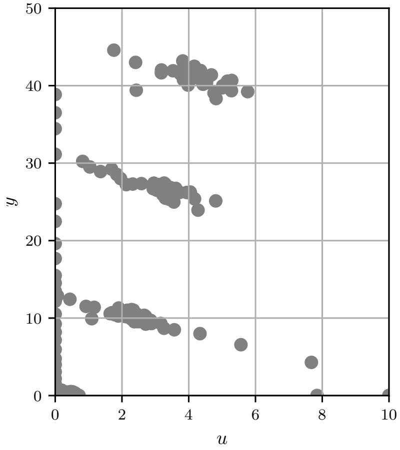
Limitations of Data-driven Outlier Detection

- need for training data
- changes in operation regime



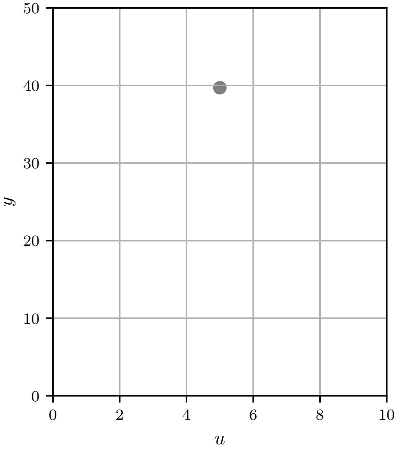
Limitations of Data-driven Outlier Detection

- need for training data
- changes in operation regime
- unavailability of labels



Limitations of Data-driven Outlier Detection

- need for training data
- changes in operation regime
- unavailability of labels
- no storage of historical data

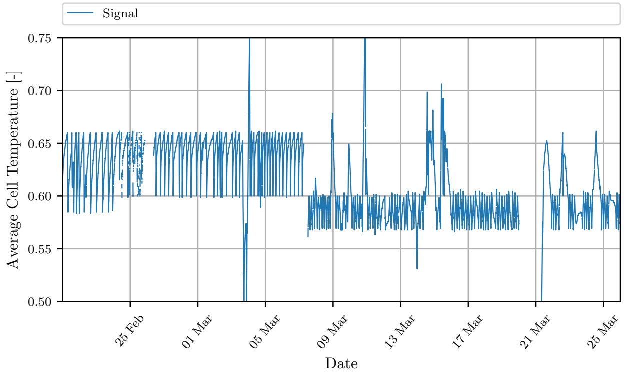


Proposed Solution

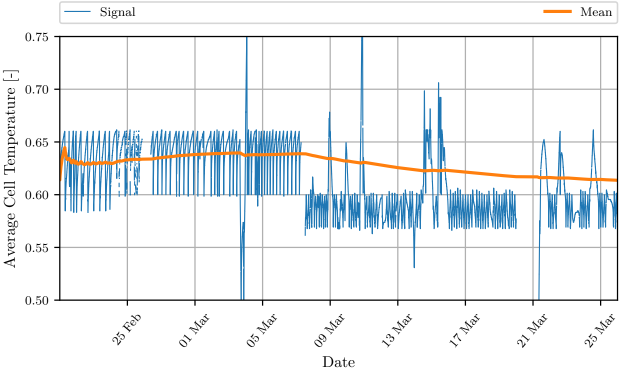
Real-Time Outlier Detection with Dynamic Process Limits combining:

- online learning
- invertible probabilistic model
- outlier detection
- self-supervised learning

Real Operation Data

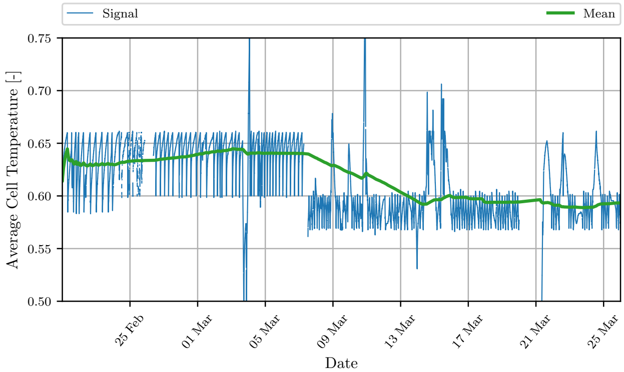


Online Learning via Welford Algorithm



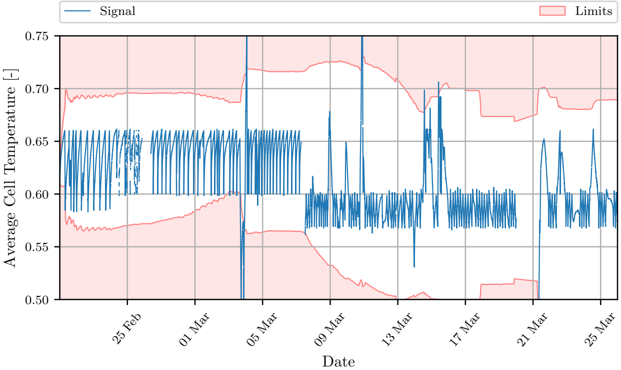
+ One-Pass Algorithm | − Adaptation Slows Down

Online Learning via Invertible Welford Algorithm



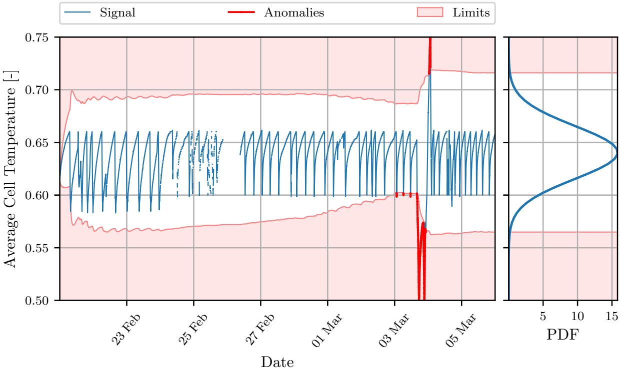
+ Constant Adaptation | − Memorizes Data Window

Dynamic Threshold Limits via Inversion



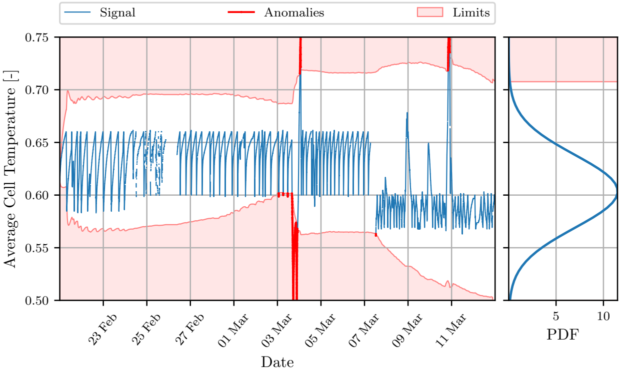
$$x_{\mathrm{u}} = F_X(q; \bar{x}_n, s_n)^{-1}$$
$$x_{\mathrm{l}} = F_X(1-q; \bar{x}_n, s_n)^{-1}$$

Distance-based Outlier Detection



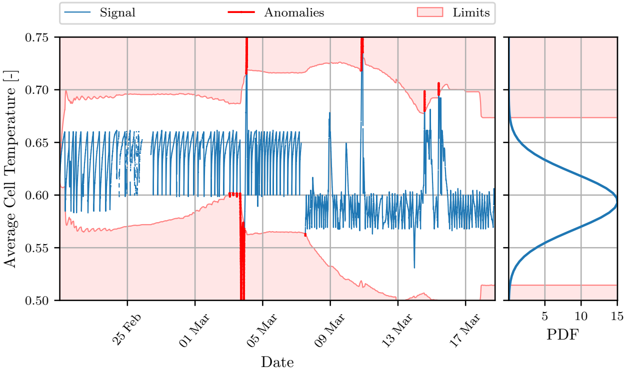
$$y_i = \begin{cases} 0 & \text{if } q \leq F_X(x_i; \bar{x}_n, s_n) \\ 1 & \text{if } q > F_X(x_i; \bar{x}_n, s_n) \end{cases}$$

Distance-based Outlier Detection



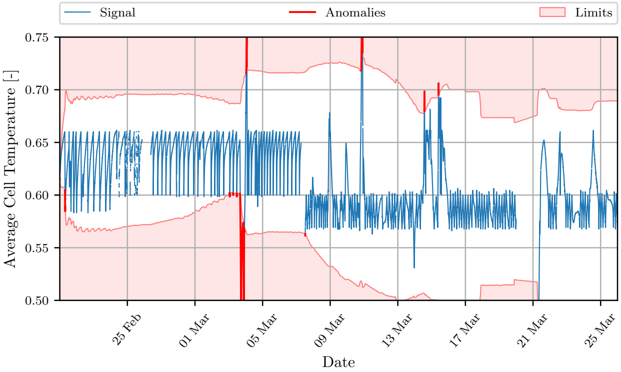
$$y_i = \begin{cases} 0 & \text{if } q \leq F_X(x_i; \bar{x}_n, s_n) \\ 1 & \text{if } q > F_X(x_i; \bar{x}_n, s_n) \end{cases}$$

Distance-based Outlier Detection



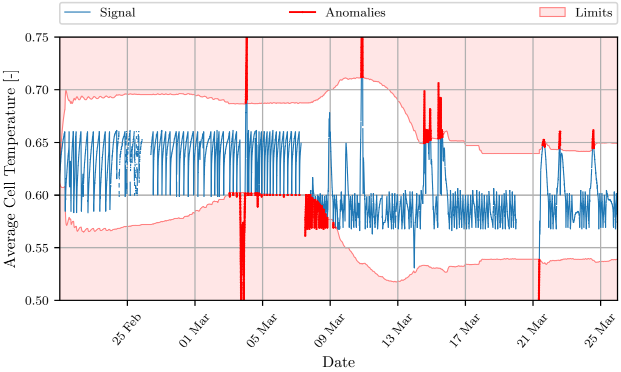
$$y_i = \begin{cases} 0 & \text{if } q \leq F_X(x_i; \bar{x}_n, s_n) \\ 1 & \text{if } q > F_X(x_i; \bar{x}_n, s_n) \end{cases}$$

Distance-based Outlier Detection



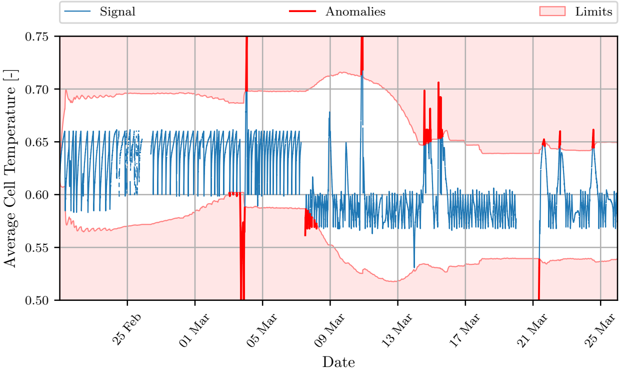
$$y_i = \begin{cases} 0 & \text{if } q \leq F_X(x_i; \bar{x}_n, s_n) \\ 1 & \text{if } q > F_X(x_i; \bar{x}_n, s_n) \end{cases}$$

Self-Supervised Learning



$$y_i = \begin{cases} 0 & \text{if } q \leq F_X(x_i; \bar{x}_n, s_n) \\ 1 & \text{if } q > F_X(x_i; \bar{x}_n, s_n) \end{cases}$$

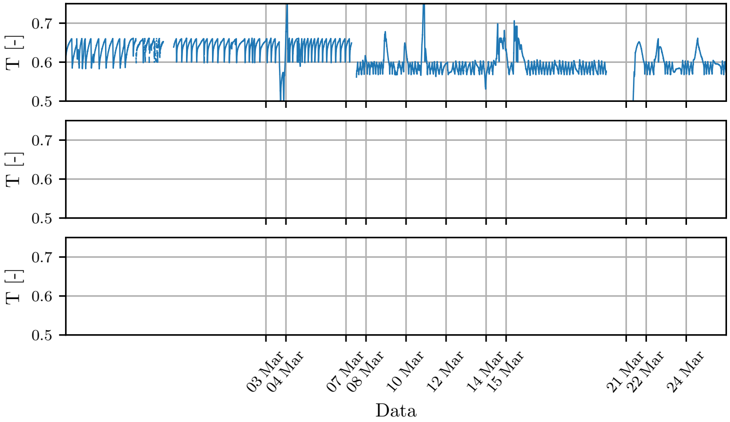
Self-Supervised Learning – Faster Adaptation



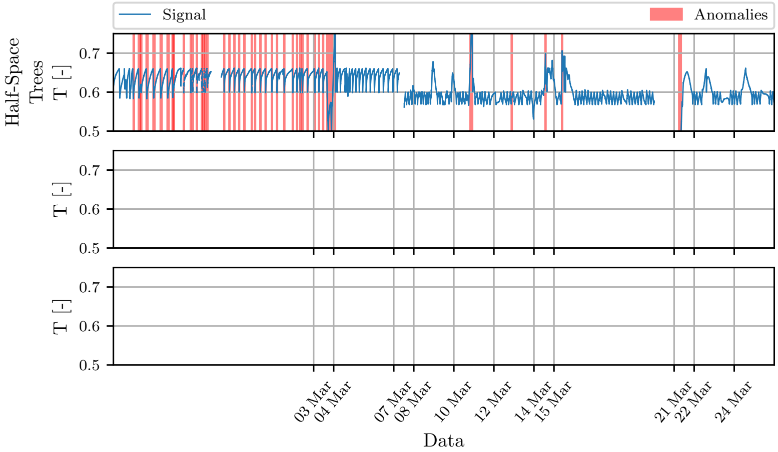
$$\frac{\sum_{y \in Y} y}{|Y|} > q$$

Battery Energy Storage System – BESS

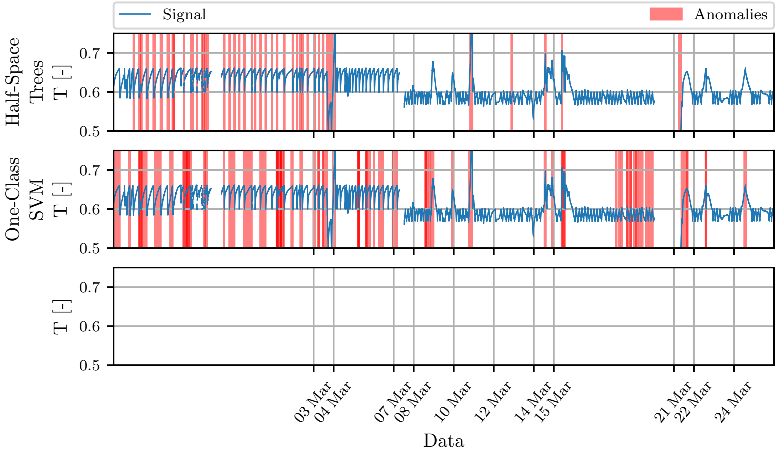
Case Study – BESS



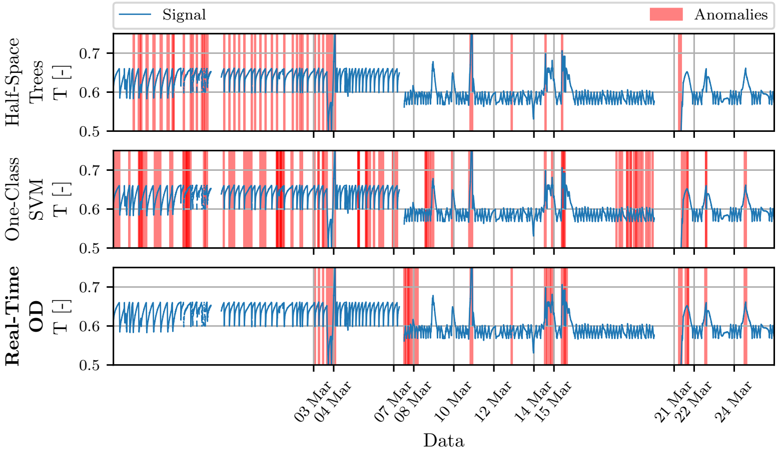
Case Study – BESS



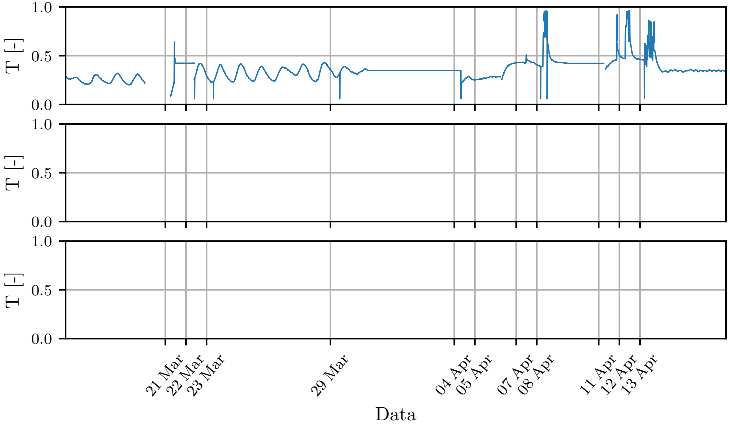
Case Study – BESS



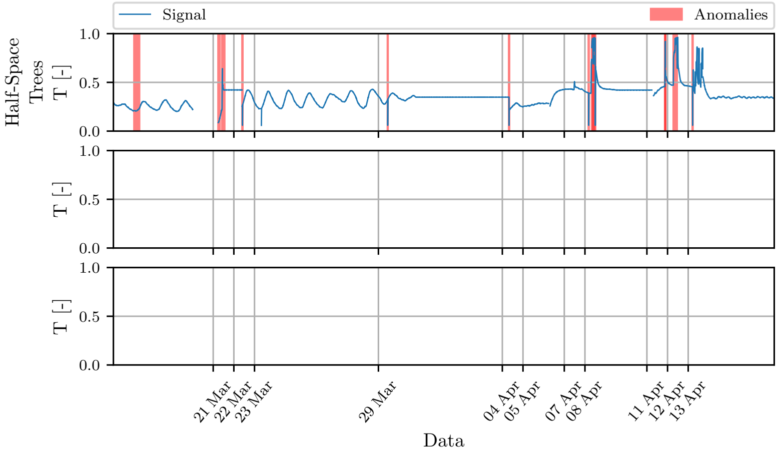
Case Study – BESS



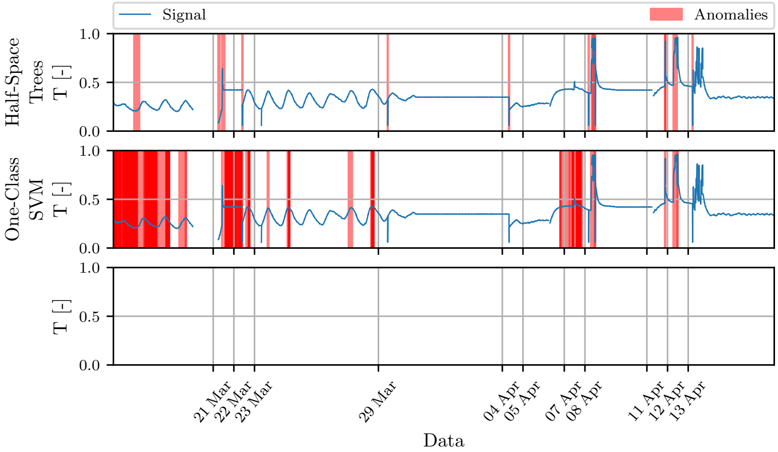
Case Study – Inverter



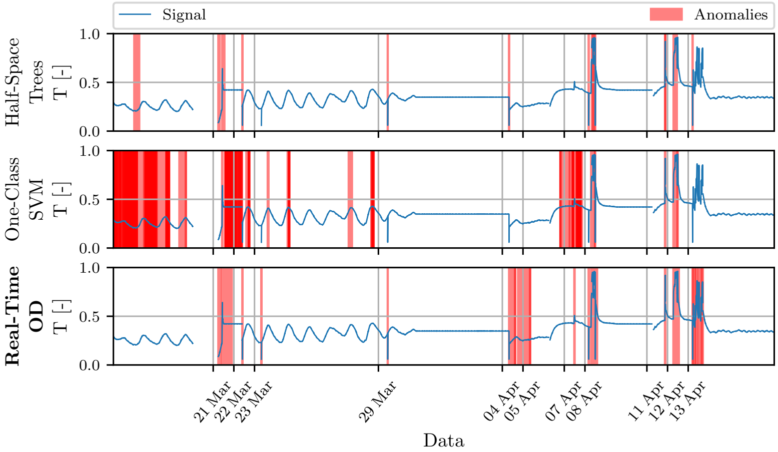
Case Study – Inverter



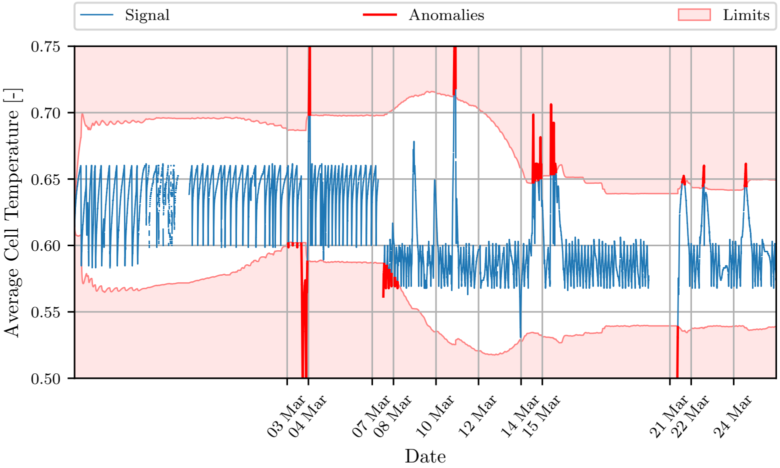
Case Study – Inverter



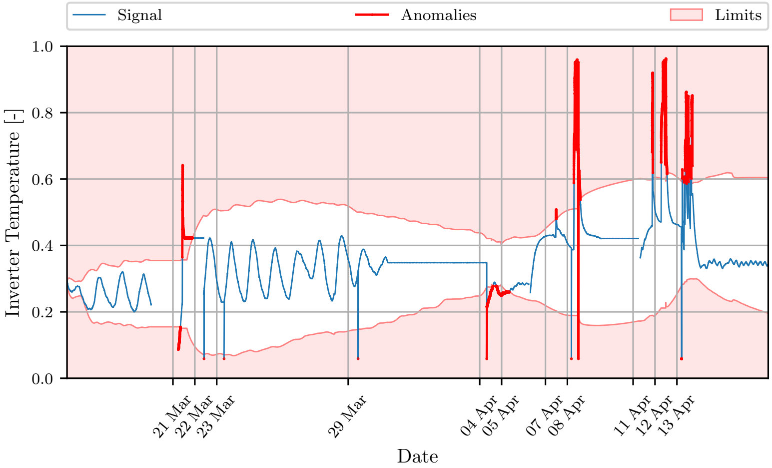
Case Study – Inverter



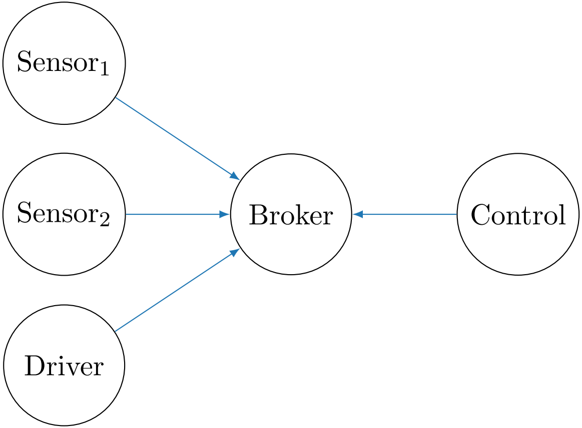
Dynamic Process Limits – BESS



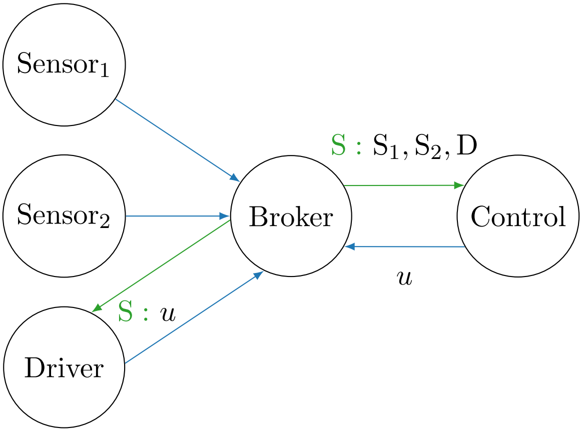
Dynamic Process Limits – Inverter



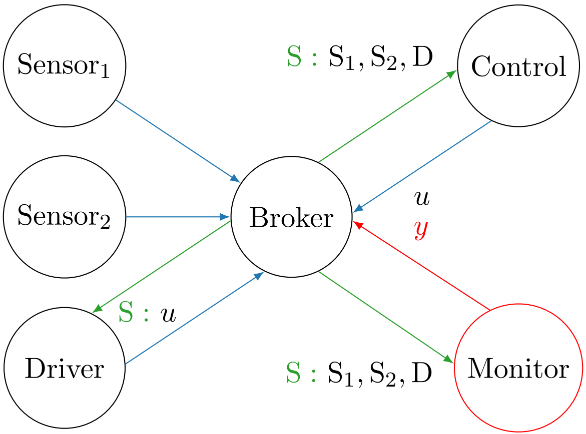
Utilize Existing Infrastructure



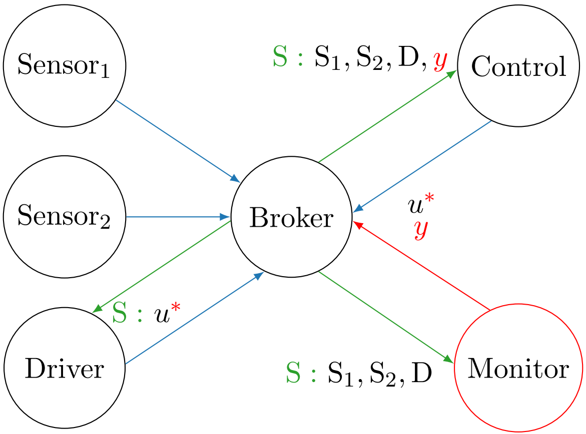
Utilize Existing Infrastructure



Utilize Existing Infrastructure

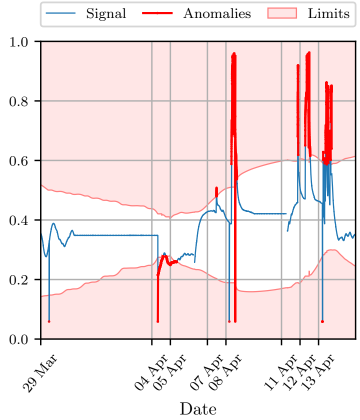


Utilize Existing Infrastructure



Summary

- outlier detection on streamed data
- adaptation to external conditions
- online self-learning approach
- dynamic process limits for individual signals
- integration with existing IT infrastructure



Acknowledgements: APVV-20-0261. VEGA 1/0490/23. Horizon Europe under grant no. 101079342 (FrontSeat).

Process Time Evaluation

Comparison of latency on univariate data.

Algorithm	BESS average [μs $\pm$ μs]	BESS max [ms]	Inverter average [μs $\pm$ μs]	Inverter max [ms]
Half-Space Trees	220 $\pm$ 200	10	220 $\pm$ 200	11
One-Class SVM	8 $\pm$ 7	1	9 $\pm$ 9	2
Proposed	57 $\pm$ 55	9	60 $\pm$ 75	12

Online Anomaly Detection Workflow

Input: expiration period t_e, time constant t_c
Output: score y_i, threshold $x_{\mathrm{q},i}$

Initialisation :

$i \leftarrow 1;\ n \leftarrow 1;\ q \leftarrow 0.9973;\ \bar{x} \leftarrow x_0;\ s^2 \leftarrow 1;$

compute $F_X(x_0)$;

LOOP Process

loop

  $x_i \leftarrow$ RECEIVE();

  $y_i \leftarrow$ PREDICT(x_i) ;

  $x_{\mathrm{q},i} \leftarrow$ GET$(q, \bar{x}, s^2)$;

  if (1a) **or** (3) **then**

    $\bar{x},\ s^2 \leftarrow$ UPDATE$(x_i, \bar{x}, s^2, n)$;

    $n \leftarrow n + 1;$

    for x_{i-t_e} **do**

      $\bar{x},\ s^2 \leftarrow$ REVERT$(x_{i-t_\mathrm{e}}, \bar{x}, s^2, n)$;

      $n \leftarrow n - 1;$

    end for

  end if

  $i \leftarrow i + 1;$

end loop

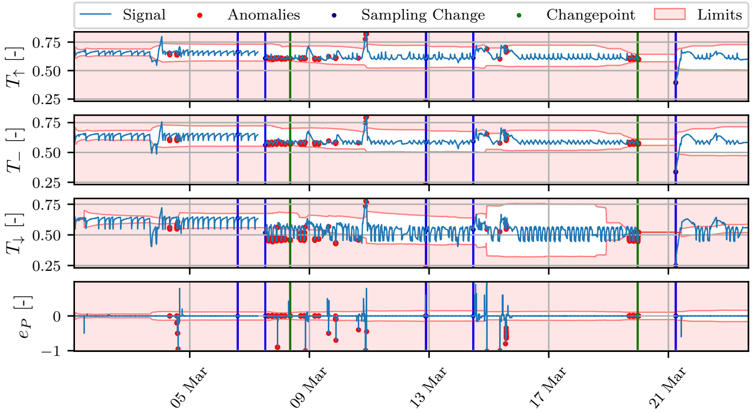
Far Less False Positive Alarms

Recall, precision, and F1 evaluation on multivariate benchmark dataset.

Algorithm	Recall [%]	Precision [%]	F1 [%]
Half-Space Trees	33	36	34
One-Class SVM	**57**	37	44
Proposed	50	**48**	**49**

M. Wadinger – M. Kvasnica. Adaptable and Interpretable Framework for Novelty Detection in Real-Time IoT Systems. IEEE CDC 2023. (reviewed)

Follow-up research



M. Wadinger – M. Kvasnica. Adaptable and Interpretable Framework for Novelty Detection in Real-Time IoT Systems. IEEE CDC 2023. (reviewed)