Real-Time Outlier Detection with Dynamic Process Limits

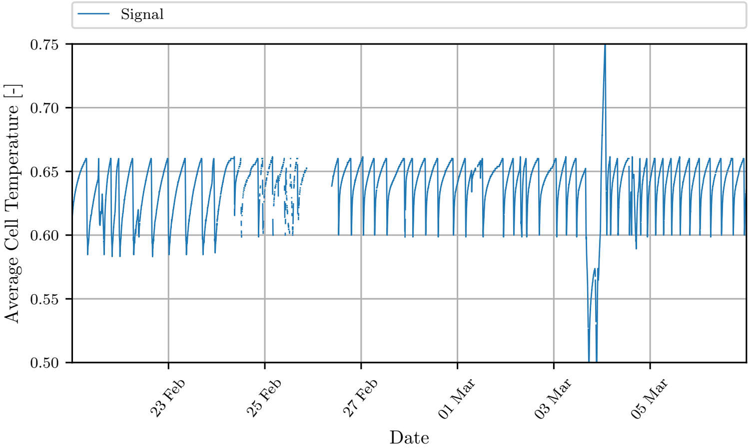
Process Control 2023

Marek Wadinger, Michal Kvasnica

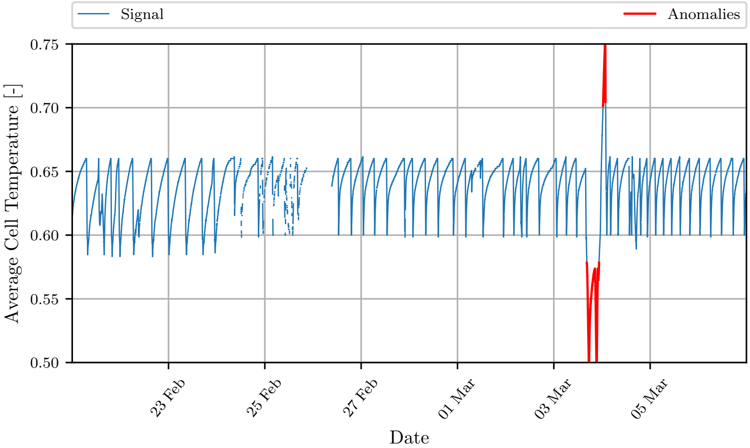
Institute of Information Engineering, Automation, and Mathematics
marek.wadinger@stuba.sk

STU FCHPT
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF CHEMICAL AND FOOD TECHNOLOGY

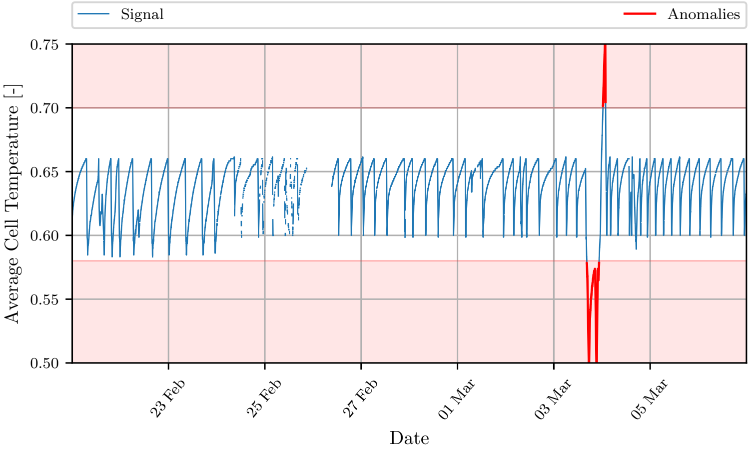
Real World Data



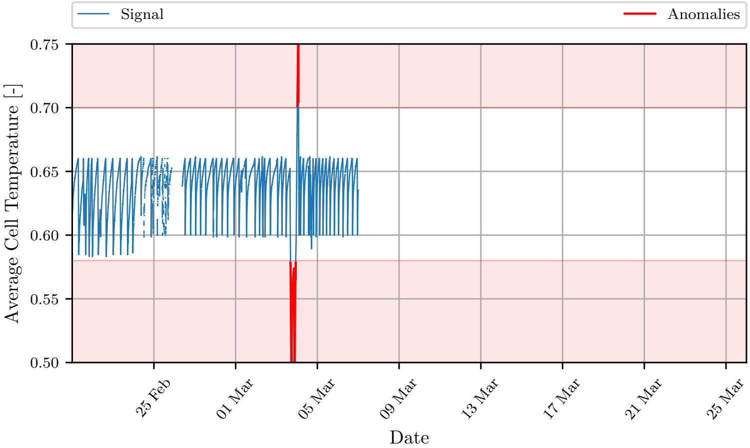
Data with Outliers



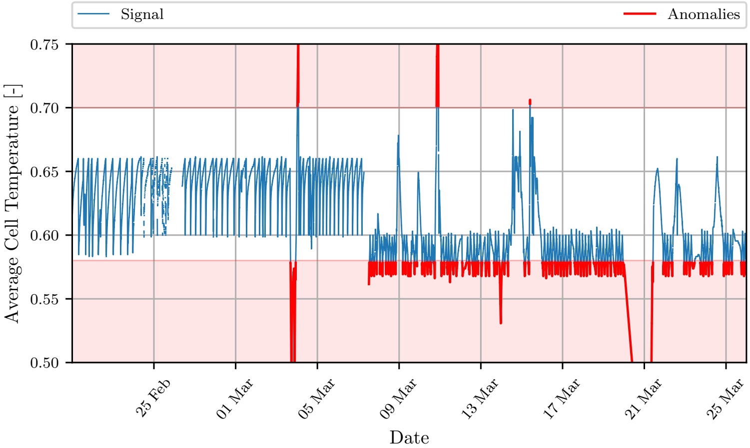
Static Threshold Limits



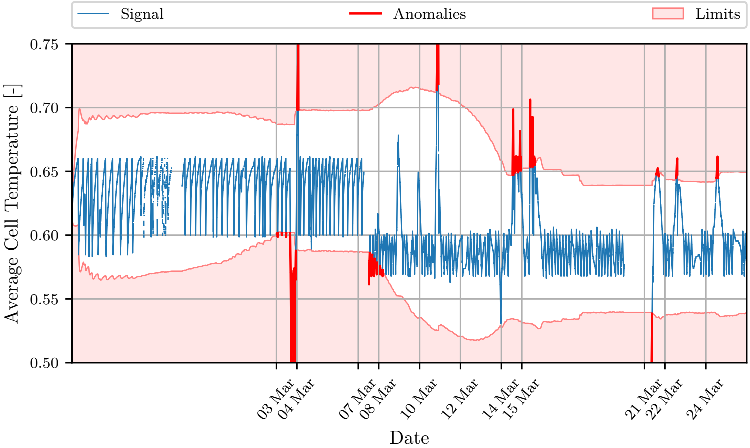
Static Threshold Limits



Comming Problem



Control Engineering Meets Artificial Intelligence



Goals

We need to design a detector that:

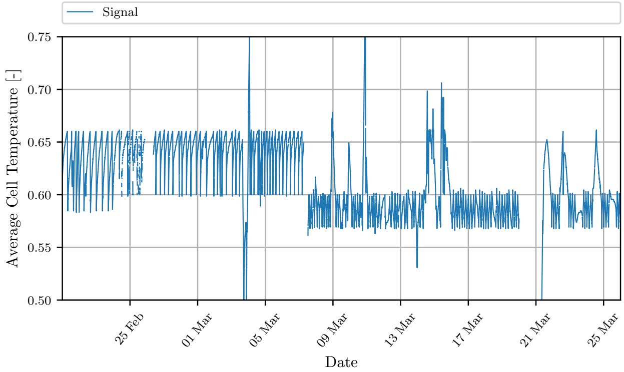
- does not require huge amount of data
- adapts to unseen operation
- provides conservative process limits
- operates with existing infrastructure

Proposed Solution

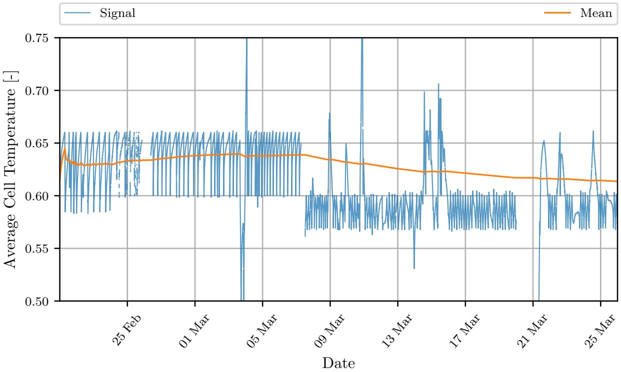
Real-Time Outlier Detection with Dynamic Process Limits combining:

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

Real Opeartion Data

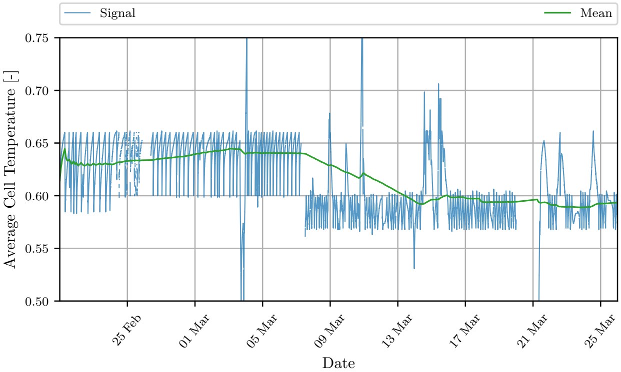


Online Learning via Welford Algorithm



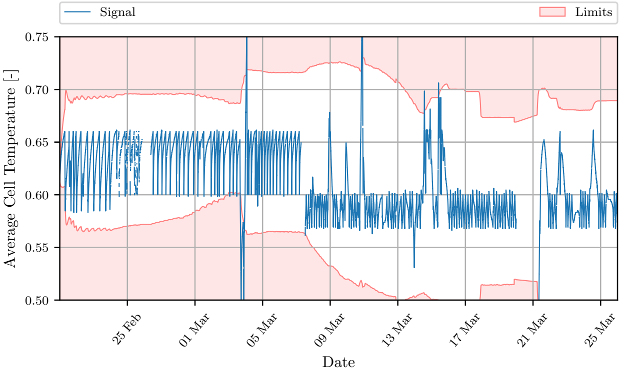
+ One-Pass Algorithm | − Adaptation Slows Down

Online Learning via Invertible Welford Algorithm



\+ Constant Adaptation | − Memorizes Data Window

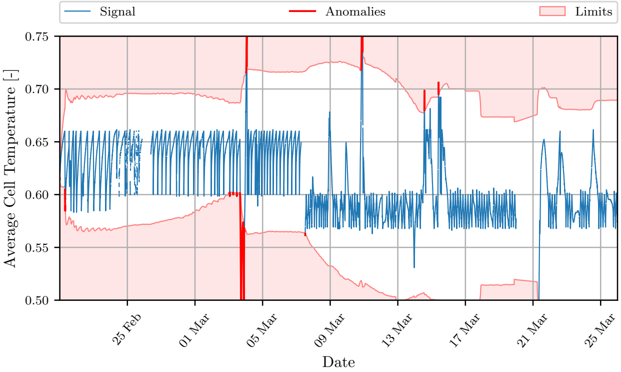
Dynamic Threshold Limits via Inversion



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

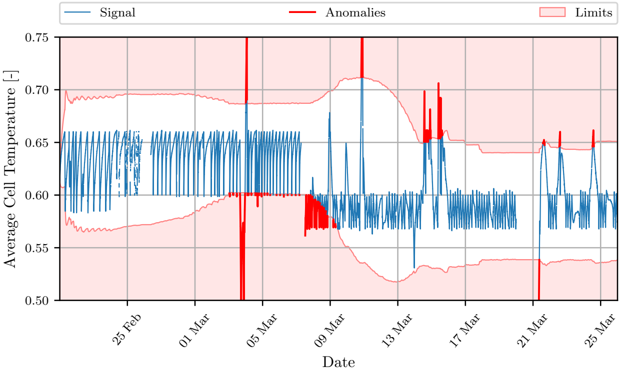
$$x_{\mathrm{u}} = F_X(q; \bar{x}_n, s_n)^{-1}$$

Distance-based Outlier Detection



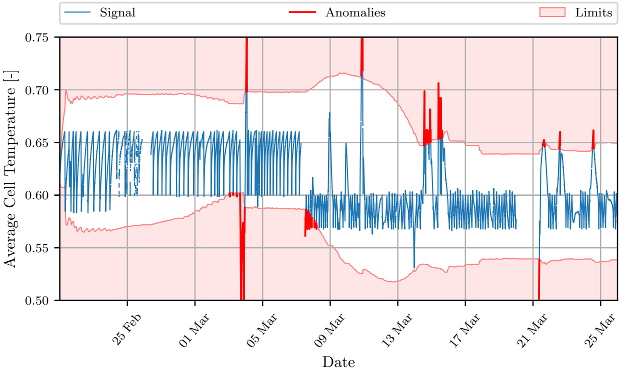
$$y_i = \begin{cases} 0 & \text{if } q \leq F_X(x_i; \bar{x}_n, s_n) \\ 1 & \text{if } q > F_X(x_i; \bar{x}_n, s_n) \end{cases}$$

Self-Supervised Learning



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

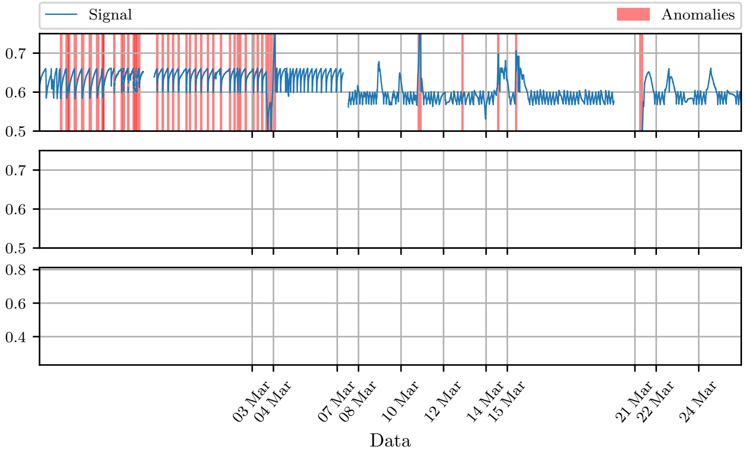
Self-Supervised Learning



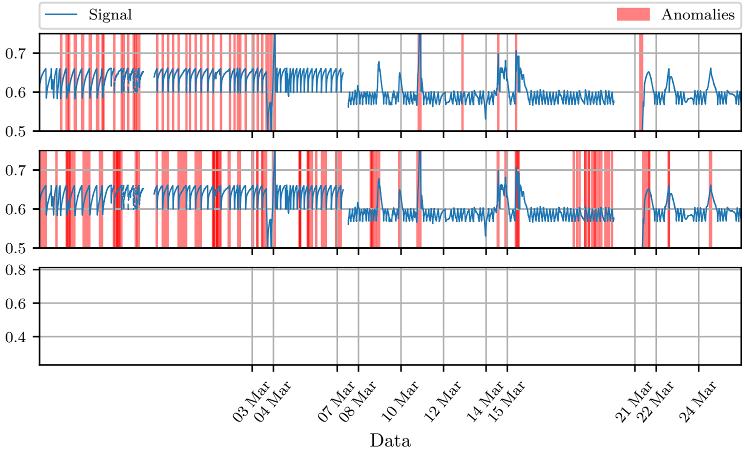
$$\frac{\sum_{y \in Y} y}{|Y|} > q$$

Battery Energy Storage System - BESS

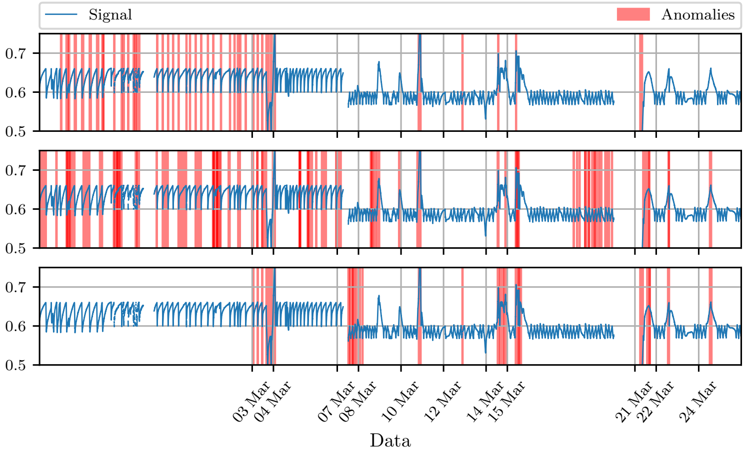
Case Study - BESS



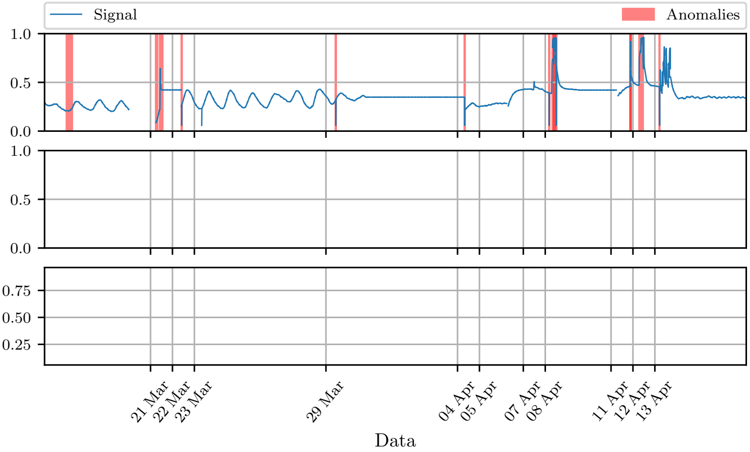
Case Study - BESS



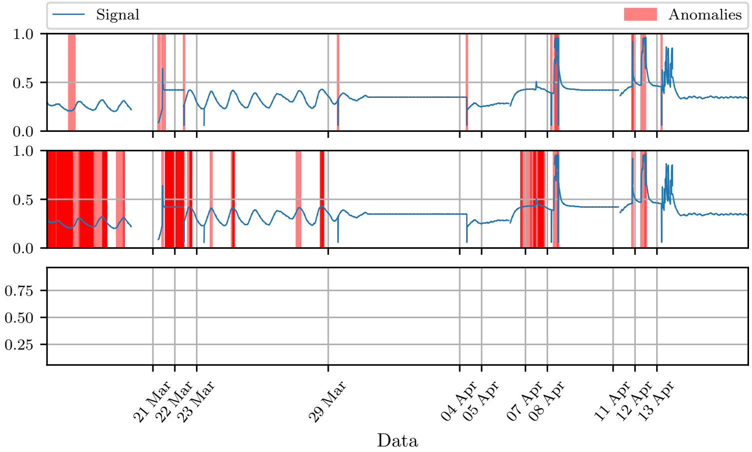
Case Study - BESS



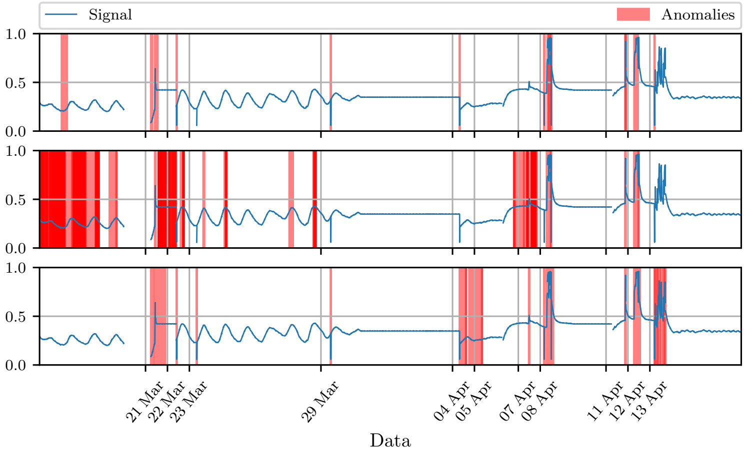
Case Study - Inverter



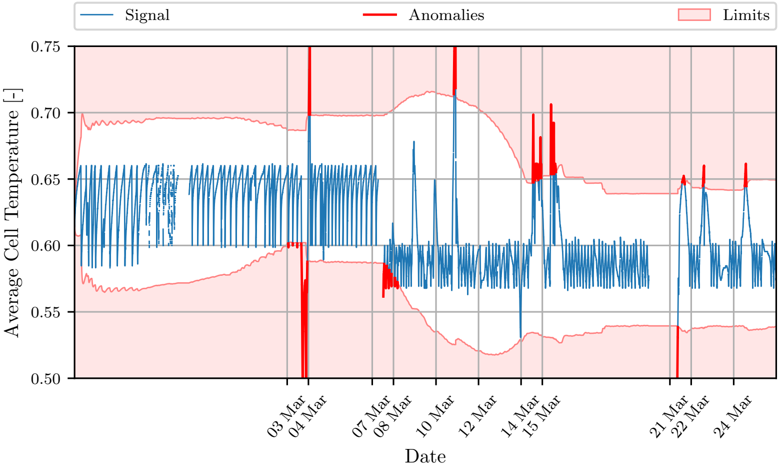
Case Study - Inverter



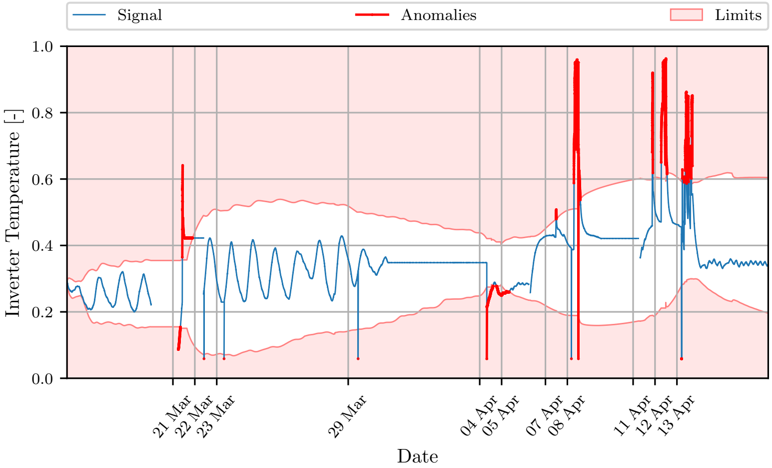
Case Study - Inverter



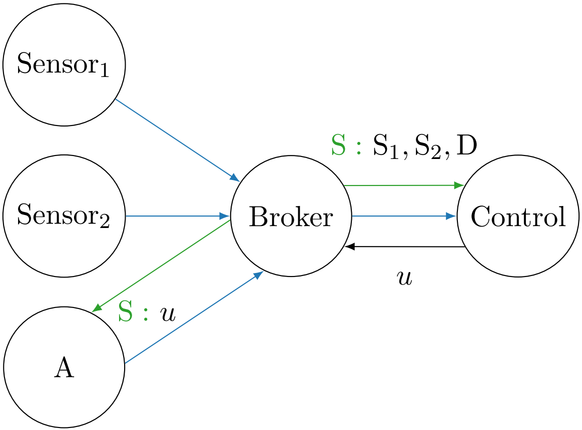
Dynamic Process Limits - BESS



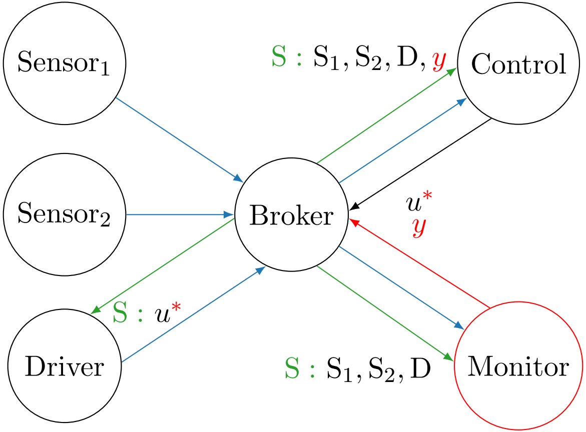
Dynamic Process Limits - Inverter



Utilize Existing Infrastructure

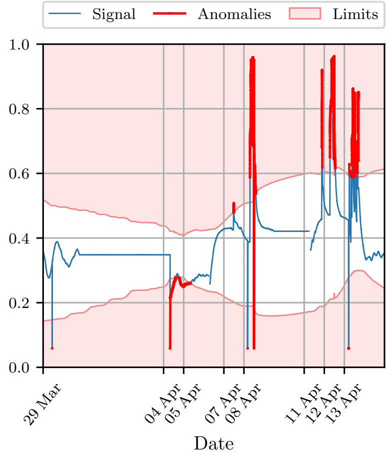


Utilize Existing Infrastructure



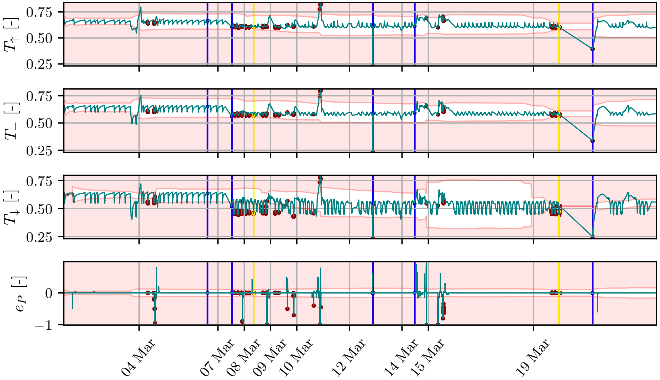
Summary

- 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 the grant no. 101079342.

Follow-up research



[1]

[1] M. Wadinger and M. Kvasnica. Adaptable and interpretable framework for novelty detection in real-time iot systems. In Proceedings of the 62nd IEEE CDC, Singapore, 2023. under review.

Process Time Evaluation

Evaluation Time [μs]	BESS		Inverter	
	average	max	average	max
Half-Space Trees	218 ± 198	10129	216 ± 196	11281
One-Class SVM	8 ± 7	1233	9 ± 9	1711
Proposed	57 ± 55	9431	60 ± 75	12330

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