

Real-Time Outlier Detection with Dynamic Process Limits

PC 2023

Marek Wadinger¹, Michal Kvasnica¹

¹Institute of Information Engineering, Automation, and Mathematics
marek.wadinger@stuba.sk

2023-05-23



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

① Motivation

② Gaps in Existing Solutions

③ Proposed Approach

④ Results

① Motivation

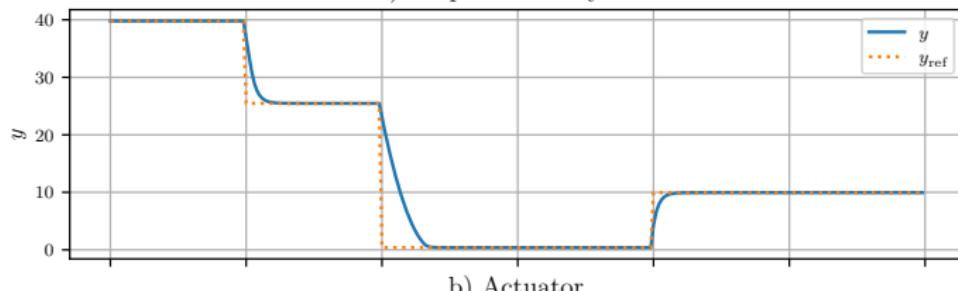
② Gaps in Existing Solutions

③ Proposed Approach

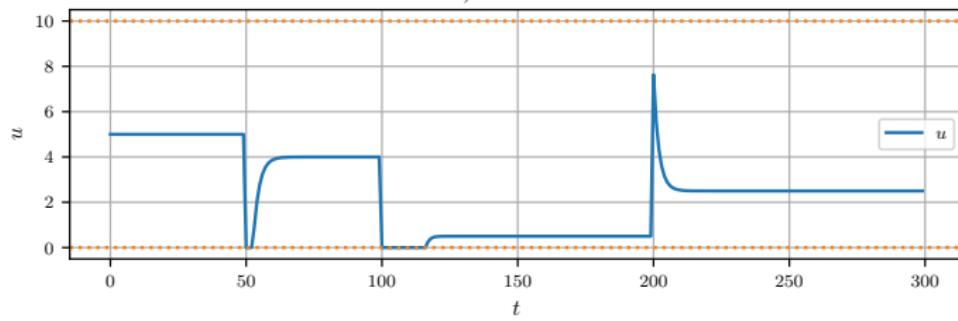
④ Results

Simulation Results

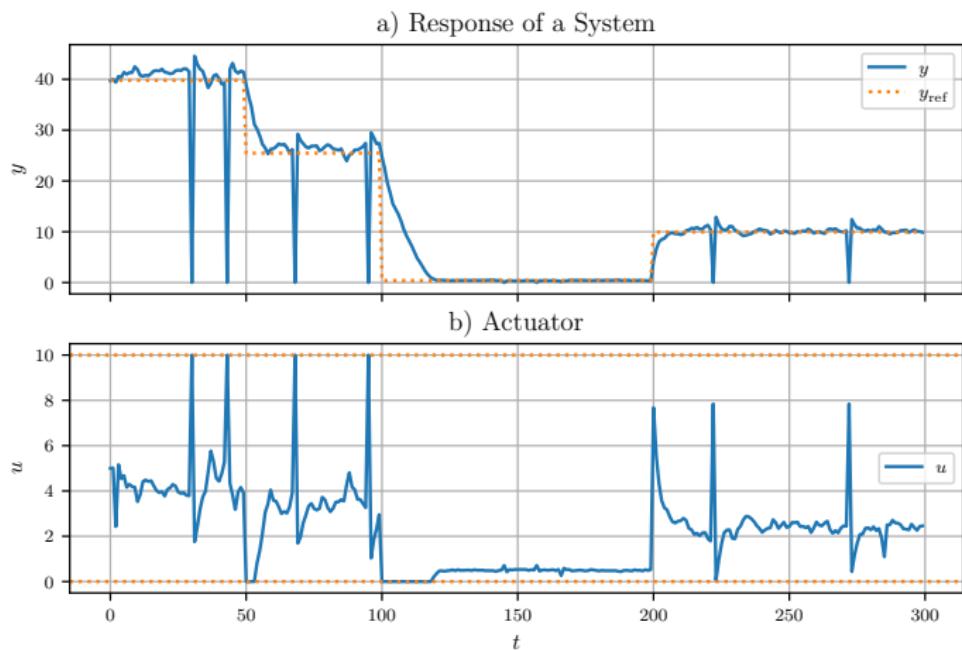
a) Response of a System



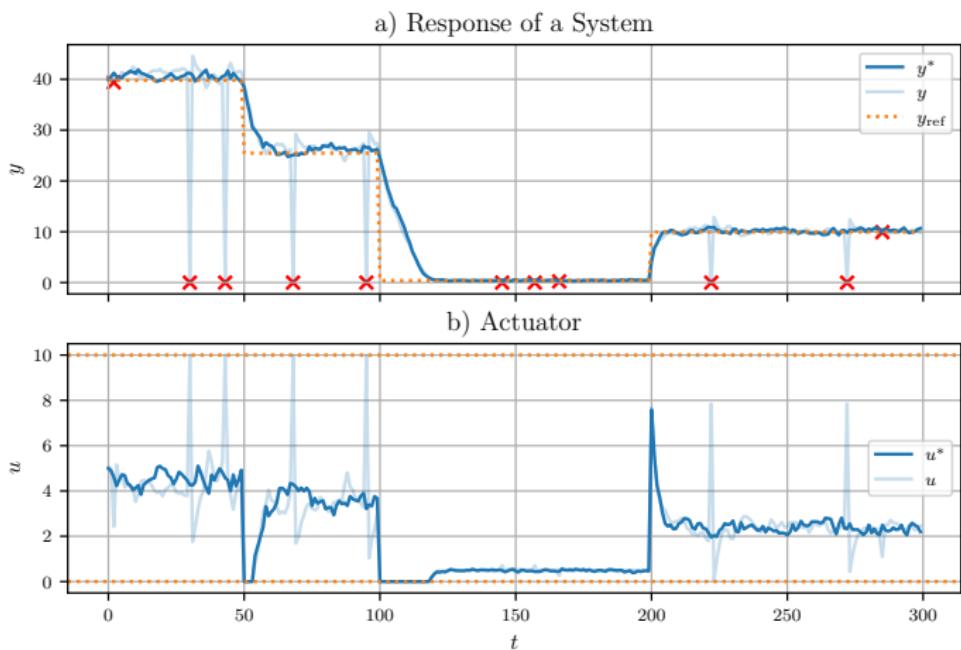
b) Actuator



Practical Scenario



Control Engineering meets Artificial Intelligence



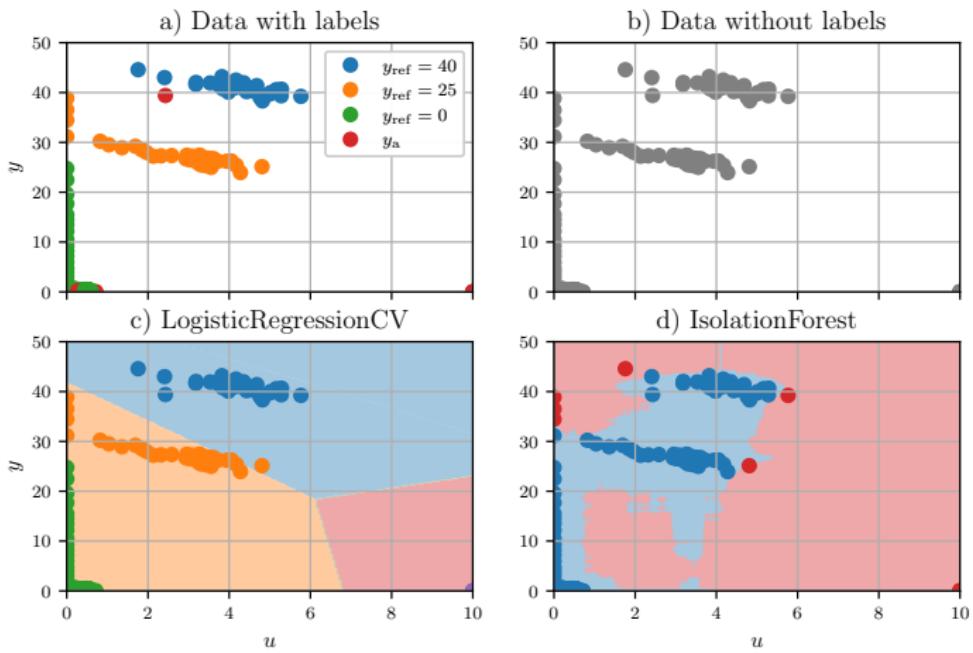
1 Motivation

2 Gaps in Existing Solutions

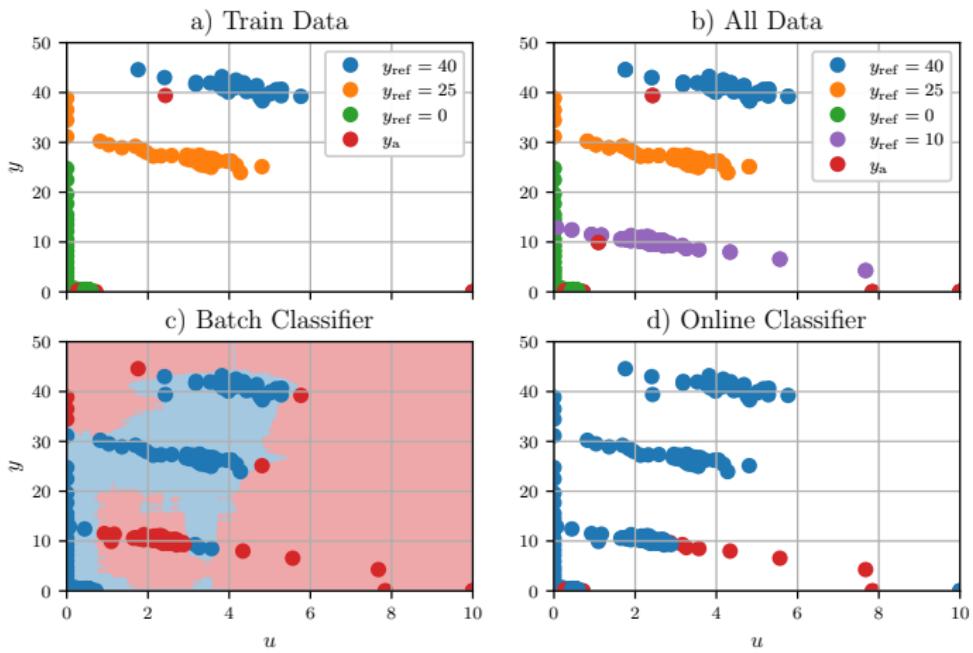
3 Proposed Approach

4 Results

Lack of Labels



Changes in Distribution



Audacious Reality

We need to make detector that:

- Does not require huge amount of data
- Adapts to unseen operation
- Offers less conservative limits
- Does not alter operation of existing systems
- Improves maintenance scheduling

① Motivation

② Gaps in Existing Solutions

③ Proposed Approach

Methodology

④ Results

Proposed Solution

Real-Time Outlier Detection with Dynamic Process Limits
combining:

- Online Learning
- Outlier Detection
- Self-supervised Learning
- Invertible Probabilistic Model

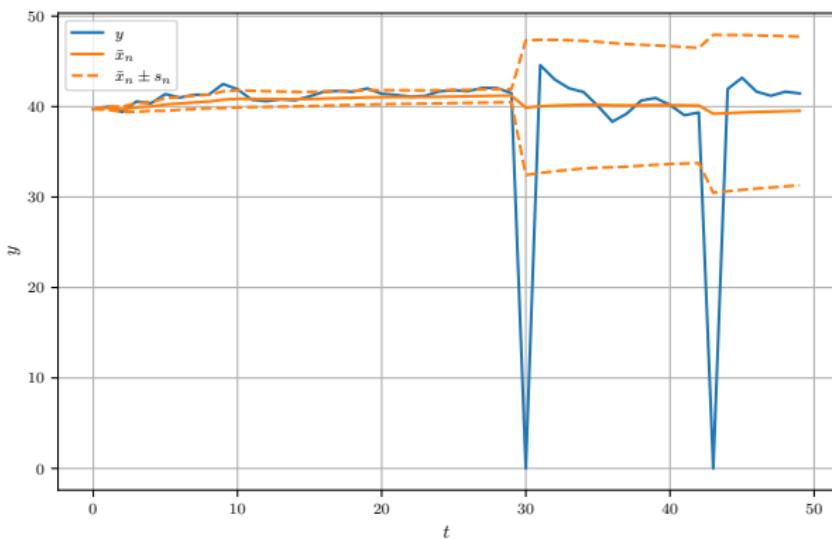
① Motivation

② Gaps in Existing Solutions

③ Proposed Approach Methodology

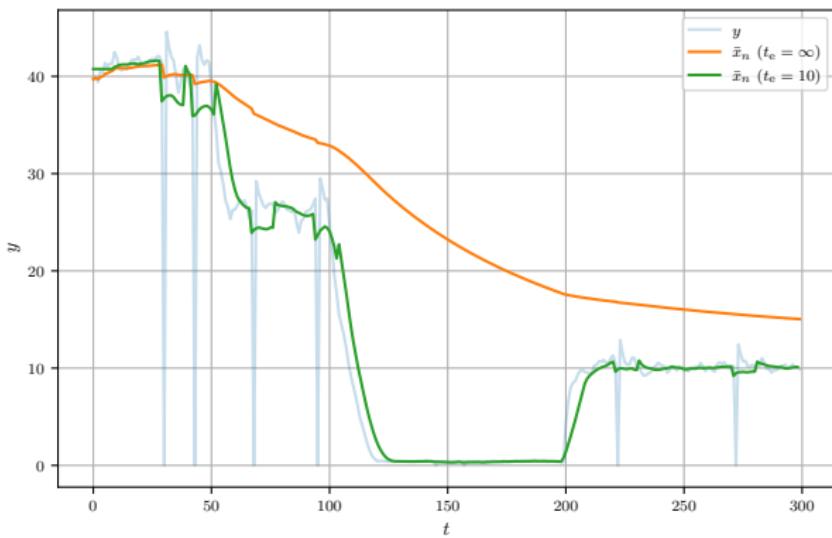
④ Results

Welford Algorithm



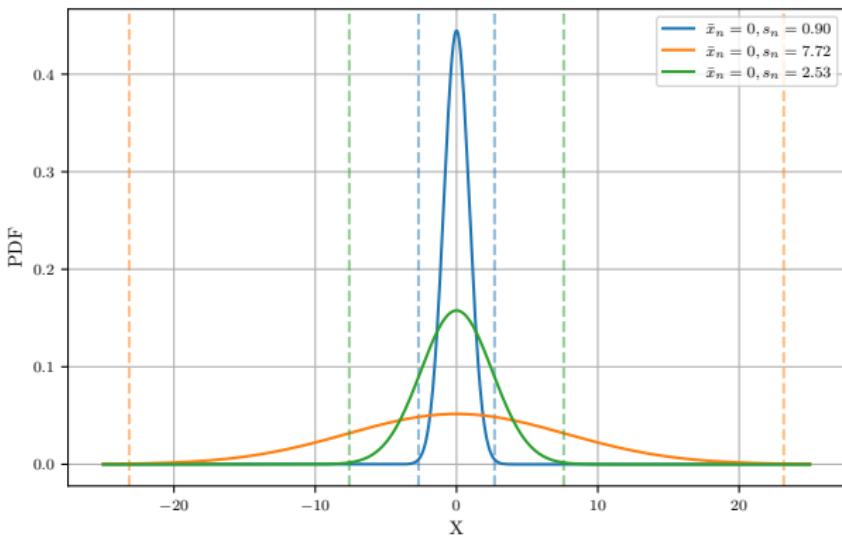
+ One-Pass Algorithm | - Adaptation Slows Down

Inverse Welford Algorithm



+ Constant Adaptation | - Memorizes Data Window

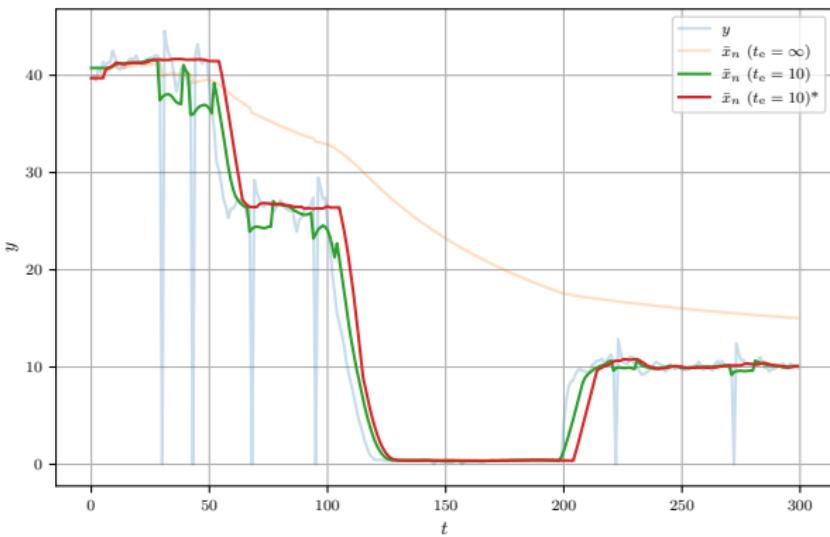
Distance-based Outlier Detection



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

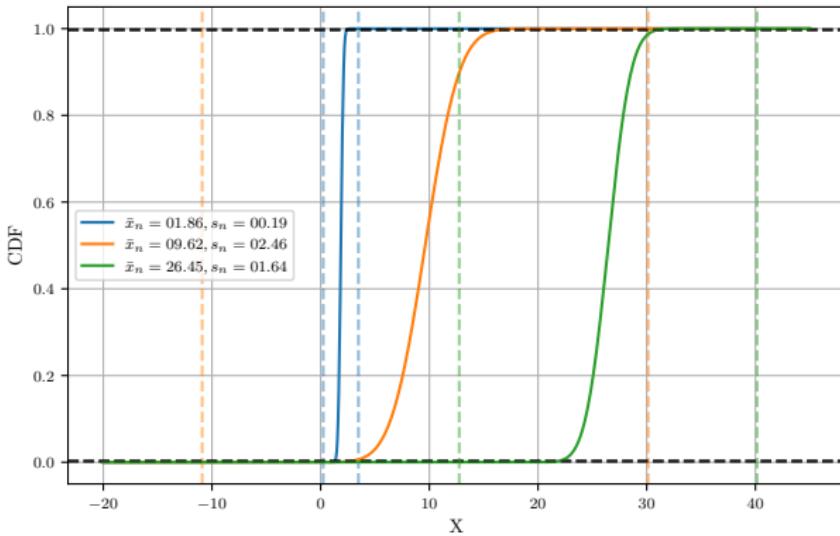
$$(1b)$$

Self-Supervised Learning



$$\frac{\sum_{y \in Y} y}{n(Y)} > q \quad (2)$$

Inversion of CDF



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

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

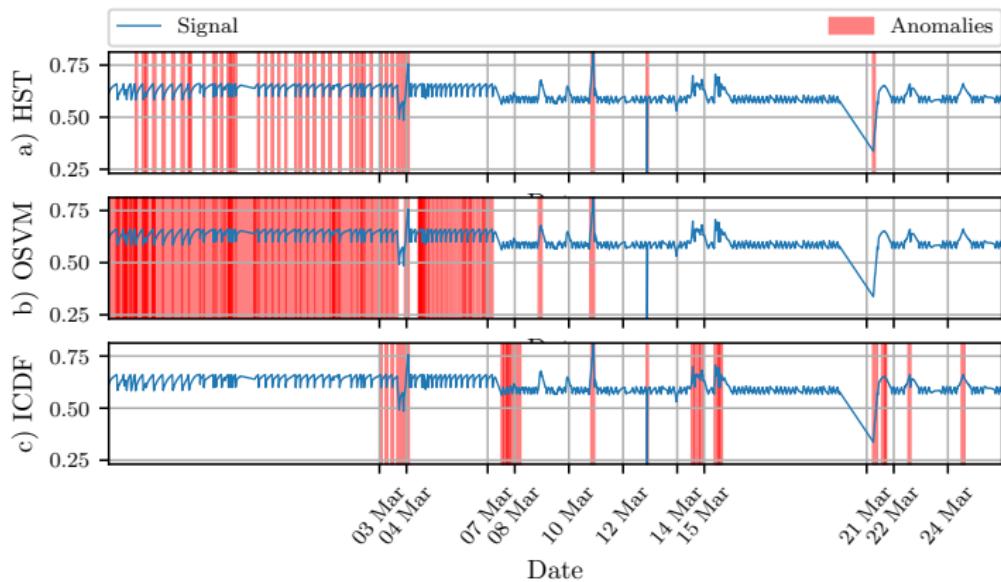
① Motivation

② Gaps in Existing Solutions

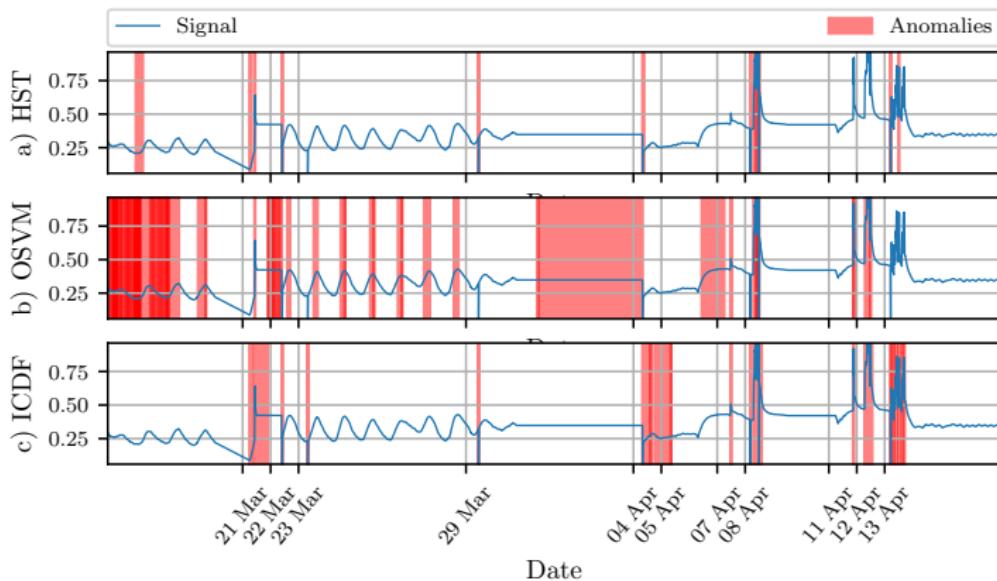
③ Proposed Approach

④ Results

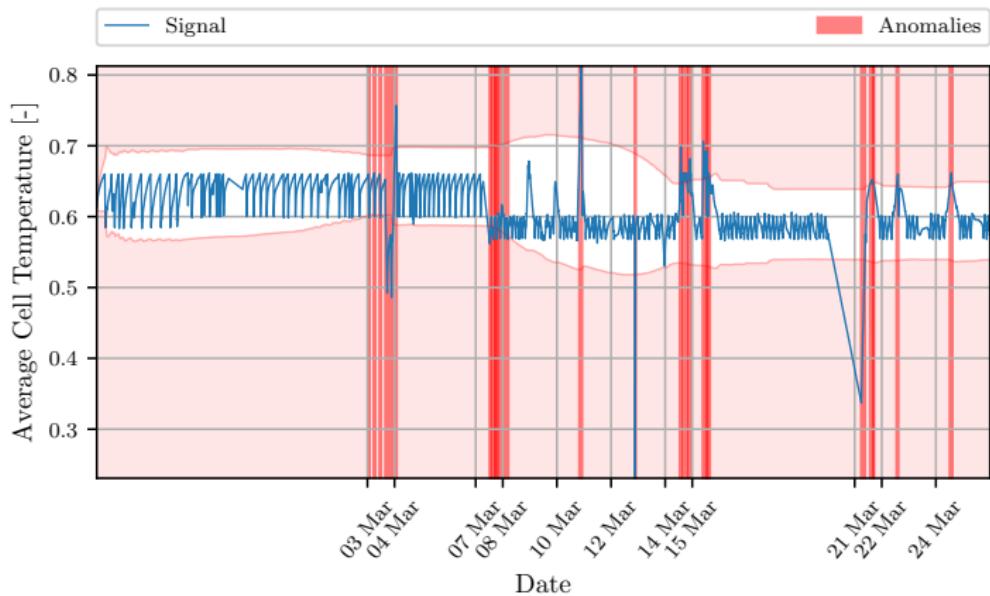
ICDF-based Outlier Detection - BESS



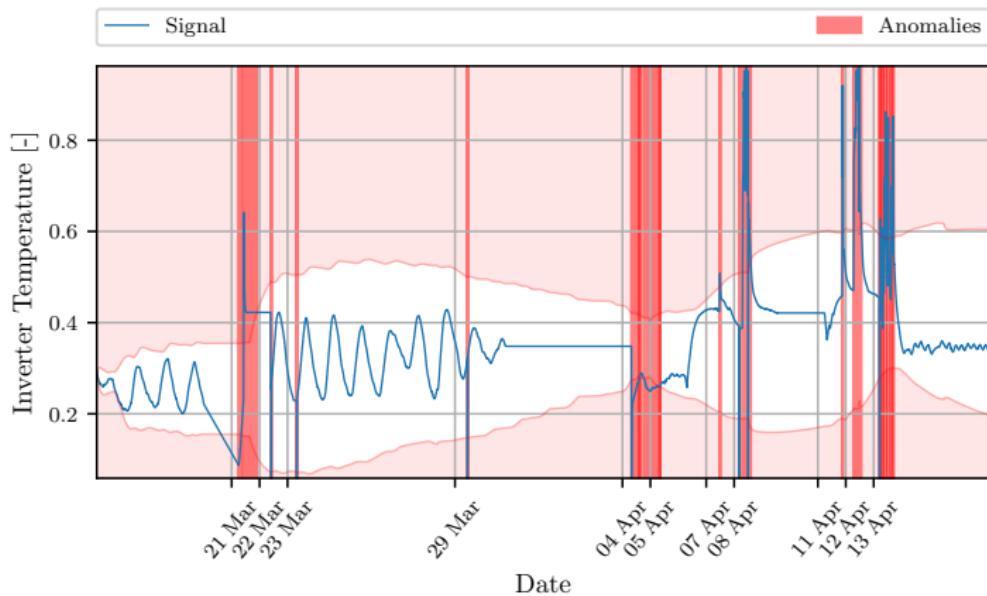
ICDF-based Outlier Detection - Inverter



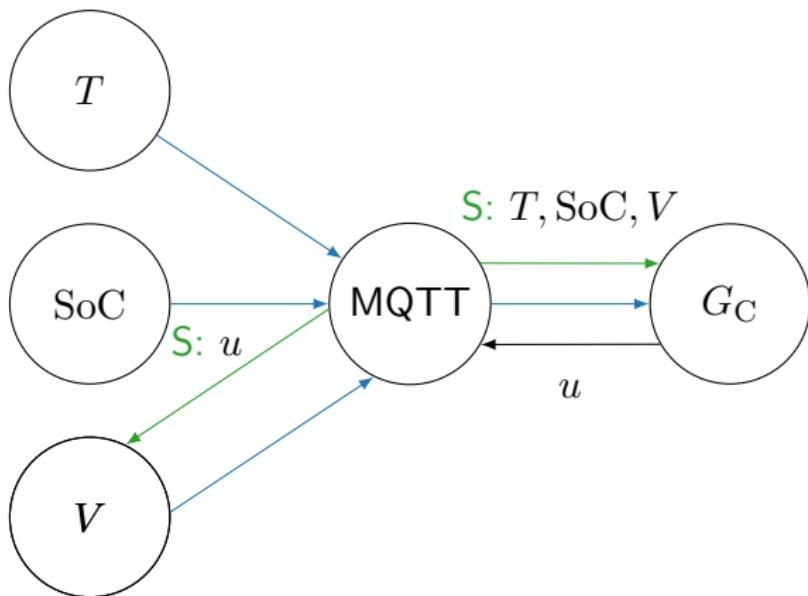
Dynamic Process Limits



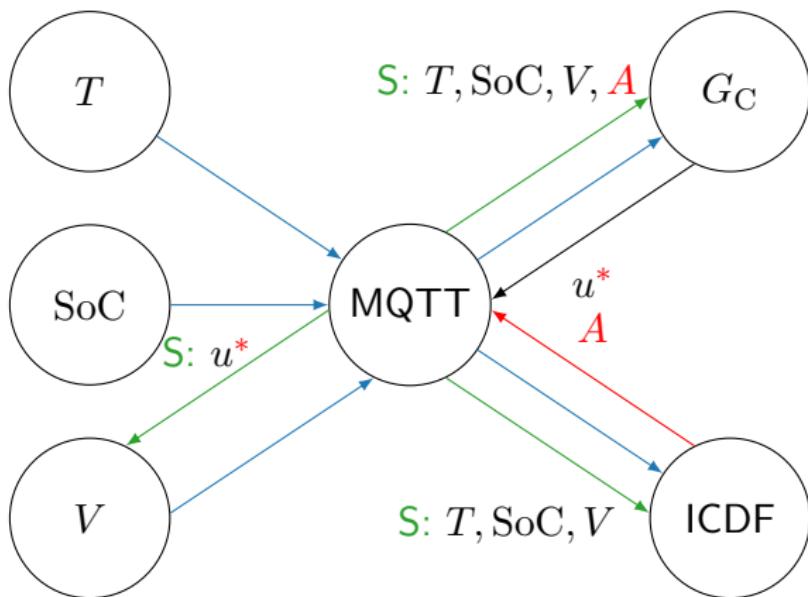
Dynamic Process Limits



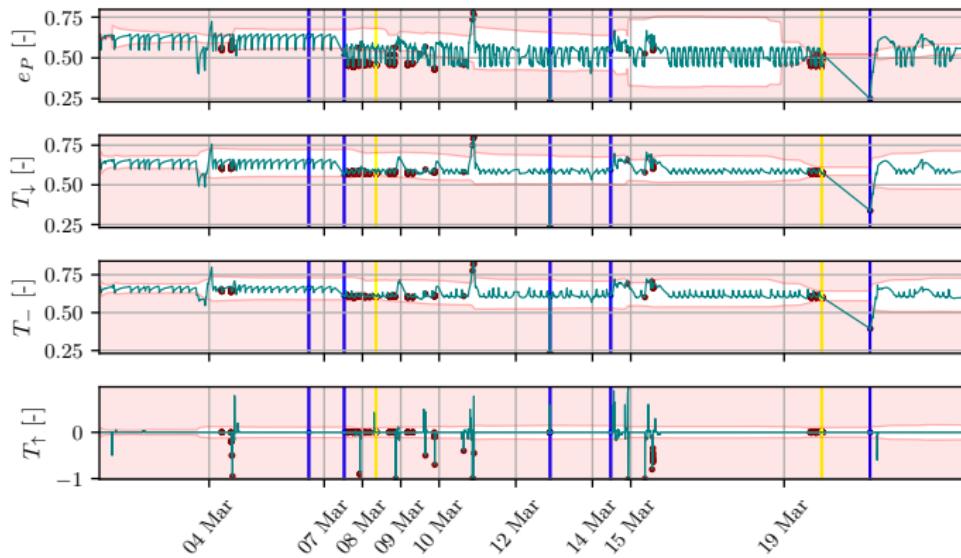
Utilize Existing Infrastructure



Utilize Existing Infrastructure



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.



Summary

- Automates alerting for signals with no a priori knowledge of process limits
- Assesses changing environmental conditions and device aging
- Uses self-learning approach on streamed data
- Automates alerting thresholds setup for a high number of signals
- Keeps existing IT infrastructure, saving costs, and does not require operator retraining

Online Anomaly Detection Workflow

Input: expiration period t_e , time constant t_c

Output: score y_i , threshold $x_{q,i}$

Initialisation :

- 1: $i \leftarrow 1; n \leftarrow 1; q \leftarrow 0.9973; \bar{x} \leftarrow x_0; s^2 \leftarrow 1;$
- 2: compute $F_X(x_0)$;

LOOP Process

3: **loop**

- 4: $x_i \leftarrow \text{RECEIVE}();$
- 5: $y_i \leftarrow \text{PREDICT}(x_i) ;$
- 6: $x_{q,i} \leftarrow \text{GET}(q, \bar{x}, s^2);$
- 7: **if** (1a) or (2) **then**
- 8: $\bar{x}, s^2 \leftarrow \text{UPDATE}(x_i, \bar{x}, s^2, n);$
- 9: $n \leftarrow n + 1;$
- 10: **for** x_{i-t_e} **do**
- 11: $\bar{x}, s^2 \leftarrow \text{REVERT}(x_{i-t_e}, \bar{x}, s^2, n);$
- 12: $n \leftarrow n - 1;$
- 13: **end for**
- 14: **end if**
- 15: $i \leftarrow i + 1;$
- 16: **end loop**