Anomaly Detection in Self-Adaptive Cyber-Physical Systems

Bachelor's Thesis/Studienarbeit

Supervisor: Prof. Dr. Alexander Pretschner
Advisor: Ana Petrovska
Email: {alexander.pretschner, ana.petrovska}@tum.de
Phone: +49 (89) 289 - 17830

Starting date: 14.11.2019

Fakultät für Informatik
Lehrstuhl 4
Software & Systems Engineering
Prof. Dr. Alexander Pretschner

Boltzmannstraße 3
85748 Garching bei München

Tel: +49 (89) 289 - 17830
https://www4.in.tum.de

Context

Cyber-Physical Systems (CPSs) are, in general, subject to unanticipated influences of their environment (e.g. unexpected behavior of the other systems in the context, or attacks by hackers), and unexpected changes in the CPSs themselves (e.g. hardware failure). Instances of negative influences correspond to anomalies, which lead to a behavior of the system or a system's component that does not match the previous "normal" behavior and previous observations. Therefore, the source of an anomaly may lie outside the system, originating from the system's context, which also includes the other systems in case of multi-agent system setups (contextual anomalies), or inside the system itself (system anomalies). If the impact of these anomalies on the behavior of the CPS is not too significant, they do not necessarily lead to immediate failure of the system as a whole, and may, therefore, go unnoticed. Still, it is desirable to detect any occurring anomaly, such as to keep the CPS in its optimal state of operation, especially as the fault or under-performance of some part of the system may indicate the eventual breakdown of the entire system.

The modern software engineering practices increasingly rely on run-time data to support decisions in continuous experimentation and self-adaptation [1]. In self-adaptive systems, the adaptation is based on the knowledge in the adaptive logic. Among the rest, knowledge constituents are context and system models. On the other side, the biggest issue in data-driven modeling is that the real-time data collected by the running CPSs may contain uncertain or even wrong values, for example, due to a faulty sensor that continuously "senses" wrong information. These values can later bias the result of data analysis and influence the model's accuracy. As a result, data-driven modeling should include the detection and handling of anomalies as an essential part of the process [1]. However, anomaly detection is rarely used in any run-time decision-making or data-driven modeling approaches for self-adaptation.

Goal

The goal of this thesis is to extend the current state of anomaly detection in self-adaptive CPSs systems by:

1) First, determine which observable parameters or features of the simulated multi-robot CPS have to be tracked, and design what data needs to be collected and logged, in order to diagnose an anomalous behavior of the system.
2) Second, apply a generic framework for tunable anomaly detection proposed in [1] in a time-series dataset from the robotics domain, considering a multi-robot scenario, with a special focus on anomalies originating from hardware faults, in particular sensor failures, and
3) And finally, the current framework supports only anomaly detection for a univariate time-series, and we want to extend the existing framework, so it can be used for detecting anomalies on multivariate time-series instead, considering that we will collect data from multiple robots

The sensor noise cannot be accessed directly. Rather, the anomalous behavior that results from the faulty sensors manifests itself indirectly in observable features like e.g. an increased number of collisions. It is evident that some observable features reflect the increased sensor noise more than others. Additionally, it can also happen that detecting an anomaly on a single time-series does not suffice, and instead a combination of multiple observables (from different robots or CPSs) is necessary to be able to detect the anomaly.

As a very last step in this thesis, we want to amend the previously built adaptation logic of the multi-robot CPS by a discriminator that makes an online decision on if a given current observation of the system's state is to be considered anomalous or not, given all observations made so far. Ideally, this is achieved by extending the approach of the framework [1], such that less expert input is needed.

Working Plan

1. Get familiar with the related work in data collection procedures
    - How do the collected datasets look like?
    - What features are being collected?
    - Assessment of how previous works prioritize the importance of the features
2. Get familiar with existing anomaly detection framework [1] for time-series data
3. Get familiar with existing anomaly detection algorithms for multivariate time-series
4. Data collection and feature selection
    - Assessment of the all the available data that can be currently collected from the simulated multi-robot system
    - Creating an initial feature set
    - Run the initial simulations, generate and collect the initial dataset
    - If necessary, analyze, cleanse and prioritize the features in the initial feature set
    - If necessary, re-run simulations, generate and collect the final dataset
5. Apply the framework [1] on the collected time-series dataset, to identify anomalies from normal behavior
6. Extend the existing framework, to be accordingly used in multivariate setup
7. Amend the system's adaptation logic by an anomaly detection module
8. Evaluate the system in different scenarios
    - Investigate how different threshold selections impact the detection of anomalies
    - Examine how weakening the feature set (by turning off some of the selected feature variables) influence the anomaly detection
    - Compare the differences between the approach in case of univariate vs. multivariate time-series anomaly detection
9. Write the thesis.

Deliverables

- Source code of the implementation.
- Technical report with comprehensive documentation of the implementation, i.e. design decision, architecture description, API description and usage instructions.
- Final report written in conformance with TUM guidelines.

References

[1] M. R. Alam, I. Gerostathopoulos, C. Prehofer, A. Attanasi, and T. Bures, "A framework for tunable anomaly detection," in *2019 IEEE International Conference on Software Architecture (ICSA)*, pp. 201–210, March 2019.

Fakultät für Informatik
Lehrstuhl 4
Software & Systems Engineering
Prof. Dr. Alexander Pretschner

Boltzmannstraße 3
85748 Garching bei München

Tel: +49 (89) 289 - 17830
https://www4.in.tum.de